

A man in a light blue shirt is seen from the side, holding a tablet computer. He is looking at the screen, which displays a software interface. The background is a blurred industrial setting with various machines and equipment.

SIEMENS

Ejemplo de aplicación 01/2016

Controlando un Robot industrial KUKA Usando un SIMATIC S7-1500

SIMATIC S7-1500 / TIA Portal V13 SP1 / KUKA.PLC mx Automatización

<https://support.industry.siemens.com/cs/ww/en/view/109482123>

Garantía y responsabilidad

Nota

Los Ejemplos de Aplicación no son vinculantes y no pretenden ser completos en cuanto a los circuitos mostrados, equipamiento y eventualidad. Los ejemplos de aplicación no representan soluciones específicas del cliente. Solo están destinados a brindar soporte para aplicaciones típicas. Usted es responsable de asegurarse de que los productos descritos se utilicen correctamente. Estos ejemplos de aplicación no le eximen de la responsabilidad de utilizar prácticas seguras en la aplicación, instalación, operación y mantenimiento. Al utilizar estos ejemplos de aplicación, reconoce que no podemos ser responsables de ningún daño / reclamo más allá de la cláusula de responsabilidad descrita. Nos reservamos el derecho a realizar cambios en estos Ejemplos de aplicación en cualquier momento sin previo aviso.

No aceptamos ninguna responsabilidad por la información contenida en este documento.

Se excluirá cualquier reclamación contra nosotros, basada en cualquier razón legal, que resulte del uso de los ejemplos, información, programas, datos de ingeniería y rendimiento, etc., descritos en este ejemplo de aplicación. Dicha exclusión no se aplicará en el caso de responsabilidad obligatoria, por ejemplo, en virtud de la Ley de Responsabilidad por Productos de Alemania ("Produkthaftungsgesetz"), en caso de dolo, negligencia grave o lesión de la vida, el cuerpo o la salud, garantía de la calidad de un producto, ocultación fraudulenta de una deficiencia o incumplimiento de obligaciones contractuales fundamentales ("wesentliche Vertragspflichten"). Los daños por incumplimiento de una obligación contractual sustancial se limitan, sin embargo, al daño previsible, típico del tipo de contrato, excepto en caso de dolo o negligencia grave o daños a la vida, el cuerpo o la salud.

Se prohíbe cualquier forma de duplicación o distribución de estos Ejemplos de aplicación o extractos de los mismos sin el consentimiento expreso de Siemens AG.

Seguridad informa- ción

Siemens ofrece productos y soluciones con funciones de seguridad industrial que respaldan la operación segura de plantas, soluciones, máquinas, equipos y / o redes. Son componentes importantes en un concepto holístico de seguridad industrial. Teniendo esto en cuenta, los productos y soluciones de Siemens se someten a un desarrollo continuo. Siemens recomienda encarecidamente que compruebe periódicamente si hay actualizaciones de productos.

Para el funcionamiento seguro de los productos y soluciones de Siemens, es necesario tomar las medidas preventivas adecuadas (por ejemplo, el concepto de protección celular) e integrar cada componente en un concepto de seguridad industrial holístico y de vanguardia. Tercero También se deben considerar los productos que pueden estar en uso. Para obtener más información sobre seguridad industrial, visite <http://www.siemens.com/industrialsecurity>.

Para mantenerse informado sobre las actualizaciones de productos a medida que ocurren, regístrese para obtener un producto: boletín específico. Para más información visite <http://support.industry.siemens.com>.

Tabla de contenido

Garantía y responsabilidad	2
1 Tarea.....	5
1.1 Introducción.....	5
1.2 Requisitos	5
2 Solución.....	6
2.1 Visión general.....	6
2.2 Funcionalidad principal disponible	7
2.3 Componentes de hardware y software	8
2.3.1 Validez.....	8
2.3.2 Componentes utilizados	8
3 Conceptos básicos	10
3.1 Diseño de un robot industrial KUKA	10
3.2 KUKA.PLC mxAutomation	11
3.2.1 Visión general.....	11
3.2.2 Biblioteca de bloques KUKA.PLC mxAutomation	12
3.2.3 Intérprete en el controlador de robot KUKA	12
3.3 Comportamiento del bloque según el estándar PLCopen	12
3.3.1 Entradas y salidas de los bloques	12
3.3.2 Ejecución encadenada de bloques de biblioteca	13
4 Generación de programas	15
4.1 Conexión del robot KUKA al SIMATIC S7	15
4.1.1 Robot como dispositivo PROFINET IO	15
4.1.2 Instalación del archivo GSDML del robot	15
4.1.3 Integración del robot en la configuración de hardware	15
4.1.4 Conexión del SIMATIC S7 al robot	17
4.1.5 Control de varios robots con un SIMATIC S7	17
4.2 Importando la biblioteca de bloques	18
4.2.1 Extrayendo la biblioteca de bloques	18
4.2.2 Abriendo la biblioteca de bloques	18
4.2.3 Transferencia de bloques al programa de usuario	18
4.3 Estructura básica del programa	19
4.3.1 Concepto	19
4.3.2 Estructura del programa	20
4.3.3 Leer datos del robot	20
4.3.4 Datos de proceso en SIMATIC S7	21
4.3.5 Escribir datos en el robot	21
4.4 Programación de funciones básicas	21
4.4.1 Encendido del robot	21
4.4.2 Inicializando el robot	23
4.4.3 Comprobación del estado del robot	24
4.4.4 Reconocimiento de mensajes de error	27
4.4.5 Leer la posición del robot	28
4.4.6 Ajuste de la anulación de velocidad	30
4.5 Programación de funciones de movimiento	30
4.5.1 Tipos de movimiento	30
4.5.2 Definición de coordenadas	31
4.5.3 Movimiento punto a punto (PTP)	32
4.5.4 Movimiento lineal.....	33
4.5.5 Movimiento circular	34
4.5.6 Función de movimiento de posicionamiento aproximado	35
4.5.7 Control de orientación	36
4.5.8 Ejes de robot de movimiento manual (JOG)	36

4.6	Ejemplo de programa resumido	38
4.6.1	Resumen completo del programa	38
4.6.2	Vista detallada del control del robot	39
4.6.3	Vista detallada del control de movimiento	40
5	Funciones adicionales	42
5.1	Control de herramienta (pinza)	42
5.1.1	Control mediante el controlador del robot	42
5.1.2	Control directo mediante SIMATIC S7	42
5.2	Seguridad integrada	43
5.2.1	Componentes	43
5.2.2	Funcionalidad	43
5.2.3	Control	44
5.3	PROFIenergía	44
5.4	Recibir mensajes de robot en texto sin formato	45
5.4.1	Principio de funcionamiento.....	45
5.4.2	Programación.....	46
6	Apéndice	48
6.1	Sistemas coordinados.....	48
6.1.1	Visión general.....	48
6.1.2	Sistema de coordenadas original del sistema (MUNDO)	48
6.1.3	Sistema de coordenadas de la base del robot (ROBROOT) ...	49
6.1.4	Sistema de coordenadas de pieza (BASE)	49
6.1.5	Sistema de coordenadas de herramienta (HERRAMIENTA)	50
6.2	Definición de coordenadas	50
6.2.1	Ejes del robot	50
6.2.2	Coordenadas del eje	51
6.2.3	Coordenadas cartesianas	51
6.2.4	Datos adicionales: "Estado" y "Giro"	51
6.2.5	Singularidades	54
6.2.6	Punto central de la herramienta (TCP)	54
7	Enlaces y literatura	55
8	Contacto.....	56
8.1	Soporte de Siemens Robotics	56
8.2	KUKA Roboter GmbH	56
8.3	Línea directa de KUKA Roboter GmbH	57
8.4	Ventas de automatización de KUKA.PLC mx	57
9	Historia.....	57

1 Tarea

1.1 Introducción

Los robots industriales son cada vez más populares. Hoy en día, se utilizan cada vez más en máquinas e instalaciones. Su sistema mecánico estandarizado está completamente desarrollado y es altamente flexible en términos de los posibles movimientos, por lo que un robot se ha convertido en la opción preferida sobre los costosos equipos mecánicos especiales. Esto permite la producción desde el tamaño de lote uno hacia arriba, incluso sin modificaciones costosas en las máquinas y plantas.

Sin embargo, desafortunadamente, el sistema de control de la planta y el controlador del robot son típicamente dos sistemas diferentes. La comunicación entre los dos controladores tiene lugar principalmente solo a nivel de bit y los programas de movimiento del robot se almacenan permanentemente en el controlador del robot y solo pueden ser llamados por el sistema de control de la planta. Por lo tanto, es relativamente difícil implementar una respuesta flexible del robot a eventos especiales de la planta.

Además, la programación del sistema de control de la planta y la del robot difieren en gran medida en su mayoría, de modo que ambos sistemas generalmente no pueden ser apoyados por una sola persona. Por tanto, los problemas de interfaz y coordinación son inevitables.

1.2 Requisitos

El objetivo es integrar completamente el control del robot y el control de movimiento en el sistema de control de la máquina y la planta para hacer que el uso de un robot industrial en una planta de producción sea más fácil y flexible.

Los siguientes requisitos se aplican a la tarea de automatización:

- El robot debe ser totalmente programable y controlable mediante el sistema de control de la máquina y la planta (PLC).

- El robot se puede operar utilizando la misma HMI del PLC o la máquina (punto único de operación).

- El diagnóstico del robot debería ser posible en su totalidad utilizando el PLC.

- Debería ser posible integrar funciones relacionadas como Safety Integrated y PROFIenergy y controlarlos mediante el PLC.

2 Solución

2.1 Visión general

El objetivo es utilizar un controlador SIMATIC S7-1500 para controlar y operar completamente un robot industrial KUKA. Para ello, se utiliza la biblioteca de bloques KUKA.PLC mxAutomation en el TIA Portal, que proporciona todos los bloques de función necesarios.

La comunicación entre el controlador SIMATIC S7-1500 y el robot industrial KUKA se realiza a través de una conexión PROFINET. Todos los comandos y la información de estado entre el controlador SIMATIC y el robot se intercambian a través de esta conexión.

Nota

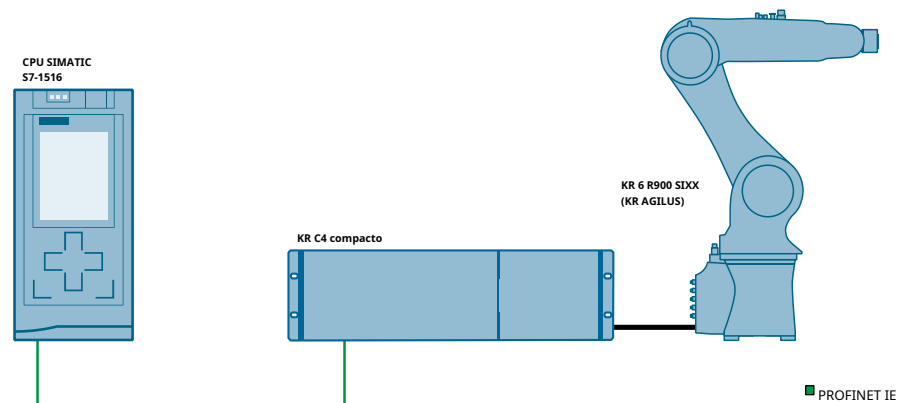
Con un controlador SIMATIC S7-1500 se pueden controlar simultáneamente hasta cinco robots industriales KUKA con la ayuda de la biblioteca de bloques KUKA.PLC mxAutomation en el TIA Portal.

Sin embargo, este ejemplo de aplicación se limita a conectar un solo robot a un SIMATIC S7-1500.

Representación esquemática

La siguiente representación esquemática muestra los componentes más importantes de la solución:

Figura 2-1 Descripción esquemática del ejemplo de aplicación



Con la biblioteca de bloques KUKA.PLC mxAutomation, el controlador SIMATIC toma el control total del robot industrial KUKA.

El robot industrial KUKA consta del controlador de robot KUKA KR C4 y el sistema mecánico del robot, el robot real. El intérprete de los comandos de la biblioteca de bloques KUKA.PLC mxAutomation está instalado en el controlador del robot. El intérprete recibe los comandos del controlador SIMATIC y los ejecuta en el sistema mecánico del robot, incluida la transformación cinemática.

Ventajas

El control de un robot industrial KUKA mediante el controlador SIMATIC S7-1500 con la ayuda de la biblioteca de bloques KUKA.PLC mxAutomation en TIA Portal ofrece las siguientes ventajas:

Solo se requiere una conexión PROFINET para conectar el robot al controlador SIMATIC.

Fácil integración del robot en TIA Portal en la configuración de hardware del controlador SIMATIC mediante un archivo GSDML.

Posibilidad de operar completamente las funciones más importantes del robot utilizando el controlador SIMATIC mediante los comandos de la biblioteca KUKA.PLC mxAutomation.

Opción para instruir movimientos del robot controlados por eventos en función del estado de los sensores conectados al controlador SIMATIC.

Operación y diagnóstico de la máquina o planta y el robot mediante una HMI común.

Alcance

El ejemplo de aplicación aquí descrito presenta un ejemplo del uso de la biblioteca de bloques KUKA.PLC mxAutomation en el TIA Portal y muestra qué funciones del robot se pueden controlar con la ayuda de los bloques de función mediante el controlador SIMATIC.

El objetivo del ejemplo de aplicación es familiarizarle con las funciones y opciones básicas de la biblioteca de bloques KUKA.PLC mxAutomation en el TIA Portal y ayudarle a tomar decisiones y planificar sus propios proyectos y programas de usuario con un robot industrial KUKA.

Para obtener una descripción detallada de las funciones y el uso de la biblioteca de bloques KUKA.PLC mxAutomation, consulte la documentación de KUKA para la biblioteca KUKA.PLC mxAutomation incluida en el capítulo "Enlaces y literatura" de esta documentación o disponible en KUKA Roboter GmbH sitio web.

Conocimiento requerido

Este ejemplo de aplicación no transmite, pero requiere conocimientos básicos sobre la creación de un programa de usuario en el SIMATIC S7-1500 en el TIA Portal o la configuración de hardware.

Además, este ejemplo de aplicación no es una introducción a la robótica. También se requieren conocimientos básicos del uso y las opciones de un robot industrial.

2.2 Funcionalidad principal disponible

A continuación se muestra una descripción general de las funciones que ofrece la biblioteca de bloques KUKA.PLC mxAutomation:

Funciones administrativas:

Encienda el robot, verifique el estado actual del robot, lea / escriba los datos de la herramienta y los datos base, use los periféricos del controlador del robot, lea / escriba los datos del interruptor de límite.

Programación de movimiento:

Mueva la cinemática del robot en líneas rectas, arcos circulares o trayectorias de velocidad optimizada. Mueva los ejes del robot en modo jog.

Control de ejecución del programa:

Interrumpa, continúe o cancele los programas del robot almacenados permanentemente en el robot controlador.

2.3 Componentes de hardware y software

Interrumpir la programación:

Defina, active / desactive las interrupciones en el controlador del robot y compruebe el estado actual.

Acciones de conmutación relacionadas con la ruta:

Defina y active condiciones de disparo para funciones de conmutación relacionadas con la trayectoria o la distancia.

Funciones de diagnóstico:

Lea los mensajes de error, las señales de diagnóstico y el estado del robot.

Funciones especiales:

Enseñe y almacene los puntos de control en el controlador del robot, realice una prueba de rotura y lea / escriba los valores del sistema.

AVISO**Uso de la opción de seguridad SafeOperation de KUKA al operar el robot a través de una HMI común sin usar el panel de control KUKA smartPAD.**

Al controlar un robot industrial KUKA mediante un controlador SIMATIC S7-1500 con la ayuda de la biblioteca de bloques KUKA.PLC mxAutomation, se debe considerar la posibilidad de utilizar adicionalmente la opción de seguridad KUKA SafeOperation. Sin embargo, si desea mover el robot en modo JOG a través de una HMI común sin utilizar el panel de control KUKA smartPAD, es obligatorio utilizar la opción de seguridad KUKA SafeOperation.

Para obtener información relacionada con la opción de seguridad SafeOperation de KUKA, consulte el capítulo 5.2 de esta documentación o llame a la línea directa de KUKA Roboter GmbH.

2.3 Componentes de hardware y software

2.3.1 Validez

Esta aplicación es válida para

STEP 7 TIA Portal V13 SP1 o superior Versión

de firmware S7-1500 V1.5 o superior

KUKA.PLC mx Biblioteca de bloques de automatización versión V2.0.2 o superior

2.3.2 Componentes utilizados

Esta aplicación fue creada con los siguientes componentes:

Componentes de hardware

Tabla 2-1 Componentes de SIEMENS

Componente	No.	Número de artículo	Nota
CPU SIMATIC S7-1516	1	6ES7 516-3AN00-0AB0	Versión de firmware: V1.5

2 Solución

2.3 Componentes de hardware y software

Tabla 2-2 Componentes de KUKA

Componente	No.	Número de artículo	Nota
KR C4 compacto con firmware V8.3	1		Controlador de robot
KR 6 R900 SIXX (KR AGILUS)	1		Robot

Componentes de software

Tabla 2-3 Componentes de SIEMENS

Componente	No.	Número de artículo	Nota
TIA Portal V13			
PASO 7 Profesional	1	6ES7 822-1AE03-0YA5	Versión: V13 SP1

Tabla 2-4 Componentes de KUKA

Componente	No.	Número de artículo	Nota
KUKA.PLC mxAutomation	1	00-233-923	Versión: V2.0.2
KUKA.Profinet Dispositivo	1	00-244-976	Versión: V3.2 Le permite operar el robot como un dispositivo PROFINET.
O alternativamente: KUKA.Profinet ControladorDispositivo	1	00-244-984	Versión: V3.2 Alternativa al dispositivo KUKA.Profinet: Le permite operar el robot como un dispositivo PROFINET o como controlador.
Archivo KUKA GSD para KR C4 con firmware V8.3	1		Versión: V2.25 Fecha de creación: 08/08/13

Proyectos y archivos de muestra

La siguiente lista contiene todos los archivos y proyectos que se utilizan en este ejemplo.

Tabla 2-5

Componente	Nota
109482123_S7-1500_KUKA_mxAutomation_DOKU_v10_en.pdf	Este documento.

3.1 Diseño de un robot industrial KUKA

3

Lo esencial

El objetivo de este capítulo es describir las funciones básicas y proporcionar información básica para el uso de un robot industrial KUKA junto con la biblioteca de bloques KUKA.PLC mxAutomation.

3.1

Diseño de un robot industrial KUKA

Un robot industrial KUKA consta generalmente de los siguientes componentes.

Figura 3-1 Diseño de un robot industrial KUKA



Tabla 3-1 Diseño de un robot industrial KUKA

No.	Componente	Función
1	Manipulador	El manipulador es el sistema mecánico real del robot, es decir, la cinemática que ejecuta los comandos.
2	panel de control smartPAD	El panel de control del smartPAD le permite realizar y verificar la configuración en el controlador del robot. Asimismo, el panel de control se puede utilizar para mover el robot de forma manual y automática.
3	Cable de conexión / smartPAD	
4	Controlador de robot	El controlador del robot coordina los movimientos del robot. Este controlador calcula la coordenada transformación para los movimientos del robot y controla los motores del eje del robot. El controlador del robot también puede contener las unidades de potencia para los motores de los ejes del robot.
5	Cable de conexión / cable de datos	
6	Cable de conexión / cable de motor	

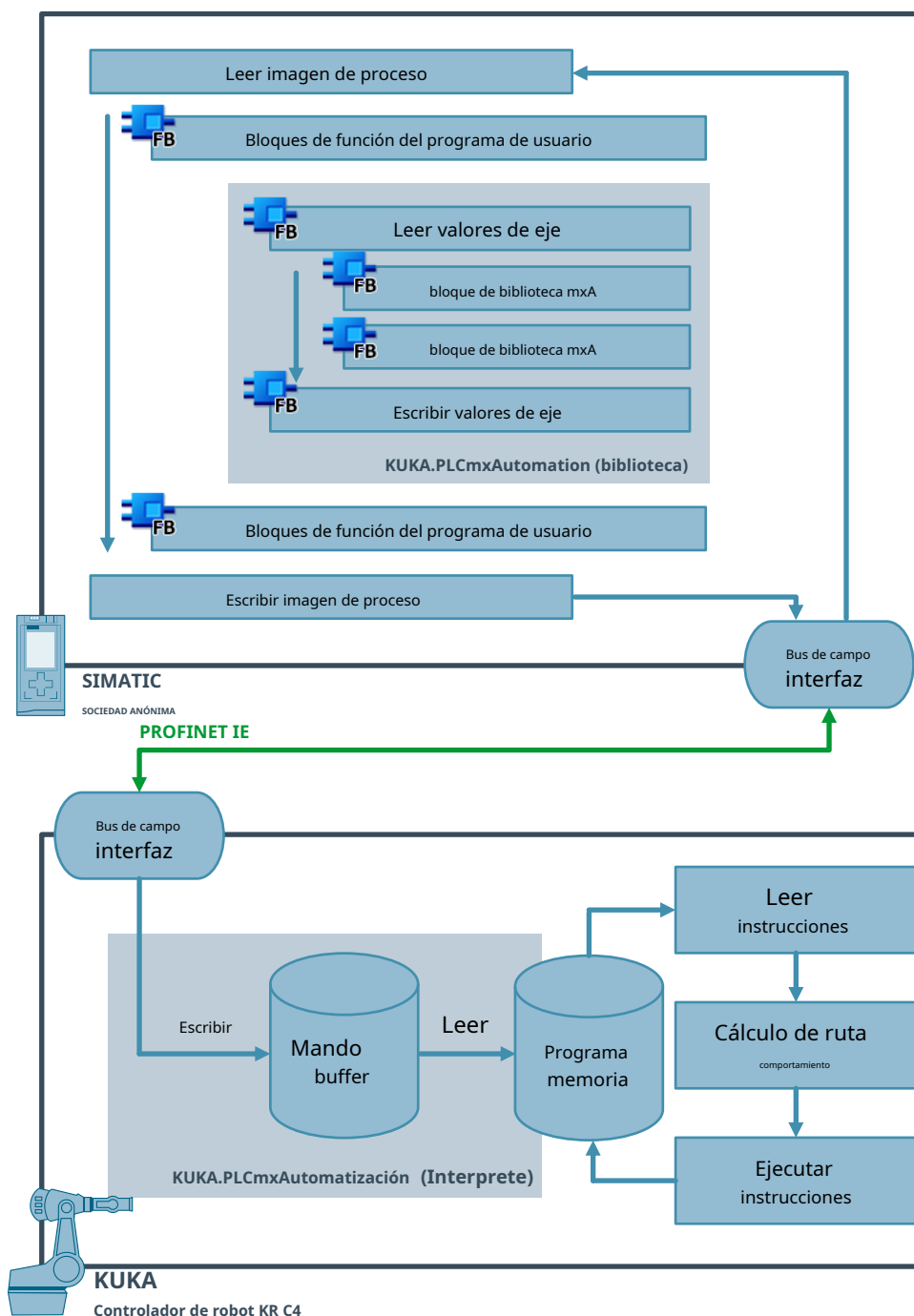
3.2 KUKA.PLC mxAutomatización

3.2 KUKA.PLC mx Automatización

3.2.1 Visión general

La siguiente figura proporciona una descripción general funcional del paquete de funciones KUKA.PLC mxAutomation.

Figura 3-2 Vista general de las funciones proporcionadas por KUKA.PLC mxAutomation



3.3 Comportamiento del bloque según el estándar PLCopen

El paquete de opciones KUKA.PLC mxAutomation para un robot industrial KUKA consta de dos partes:

Una biblioteca de bloques para controlar un robot industrial KUKA desde un controlador SIMATIC.

Un intérprete en el controlador del robot que interpreta los comandos de los bloques de función del controlador SIMATIC y los transmite a la planificación de la trayectoria del robot.

3.2.2 Biblioteca de bloques de automatización KUKA.PLC mx

La biblioteca de bloques KUKA.PLC mxAutomation ofrece diferentes bloques para controlar un robot industrial KUKA. Las funciones deseadas del robot KUKA se pueden seleccionar simplemente llamando al bloque correspondiente de la biblioteca de bloques.

La llamada de un bloque de función de la biblioteca de bloques transfiere los comandos respectivos al controlador de robot KUKA, donde se interpretan.

3.2.3 Intérprete en el controlador de robot KUKA

El intérprete del controlador de robot KUKA recibe los comandos de los bloques de función de la biblioteca de bloques KUKA.PLC mxAutomation en el controlador del robot y los almacena en una memoria intermedia. En el momento adecuado, los comandos se transfieren desde la memoria intermedia a la planificación de la trayectoria del robot para que el robot ejecute las funciones indicadas mediante el controlador SIMATIC.

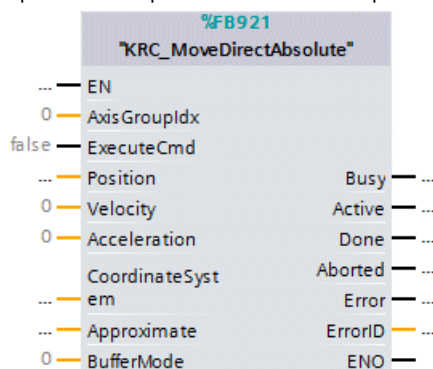
3.3 Comportamiento del bloque según el estándar PLCopen

El comportamiento de los bloques de función de la librería de bloques KUKA.PLC mxAutomation sigue el estándar PLCopen.

3.3.1 Entradas y salidas de los bloques

Los bloques de la librería de bloques KUKA.PLC mxAutomation siempre cuentan con las siguientes entradas y salidas:

Figura 3-3 Ejemplo de un bloque de la librería de bloques KUKA.PLC mxAutomation



3.3 Comportamiento del bloque según el estándar PLCopen

Tabla 3-2 Interfaces de bloque según el estándar PLCopen

Interfaz	Función
Entradas	
AxisGroupIdx	Selecciona el robot a controlar.
ExecuteCmd	Inicia la función mediante un flanco ascendente en la entrada del bloque.
Salidas	
Ocupado	La función del robot instruida a través del bloque de funciones se transfirió al controlador del robot.
Activo	El movimiento instruido a través del bloque de funciones está siendo ejecutado actualmente por el robot.
Hecho	La ejecución del movimiento por parte del robot se ha completado con éxito.
Abortado	Se canceló la ejecución del bloque de funciones o el movimiento.
Error	Se ha producido un error al ejecutar el bloque de funciones.
ErrorID	Código de error para la especificación exacta de la causa del error.

3.3.2 Ejecución encadenada de bloques de biblioteca

Con un flanco ascendente en la entrada ExecuteCmd, el bloque de función seleccionado transfiere el comando para ejecutar la función de robot deseada al controlador del robot. A través de la salida Ocupado del bloque de funciones, esta transferencia se devuelve al programa de usuario.

Si se van a ejecutar varios bloques de función de la biblioteca de bloques uno tras otro, se pueden transferir al controlador del robot en rápida sucesión. Cada vez que un bloque de funciones iniciado por un flanco ascendente establece la salida Ocupado, el comando se transfiere al controlador del robot y se puede iniciar el siguiente bloque de funciones.

Si la salida Ocupado del bloque de funciones iniciado no está configurada, la memoria intermedia en el controlador del robot está llena y el programa de usuario debe esperar hasta que el robot haya procesado los comandos o el procesamiento del bloque de funciones se canceló debido a un error o debido a la inicio de otro bloque de funciones.

Los bloques de funciones devuelven al programa de usuario los errores que se han producido a través de la salida Error. Los abortos o la sustitución del procesamiento de bloques por el inicio de otro bloque de funciones se señalizan a través de la salida cancelada.

Ejemplo de llamada encadenada de bloques de función

La siguiente figura muestra un ejemplo de la llamada de cuatro funciones de movimiento a través de los bloques de función de la biblioteca de bloques, por ejemplo, el bloque de función KRC_MoveLinearAbsolute, que se almacenan en el búfer en el controlador del robot y se ejecutan uno después del otro.

3.3 Comportamiento del bloque según el estándar PLCopen

Figura 3-4 Secuencia de señales para la llamada encadenada de bloques de función

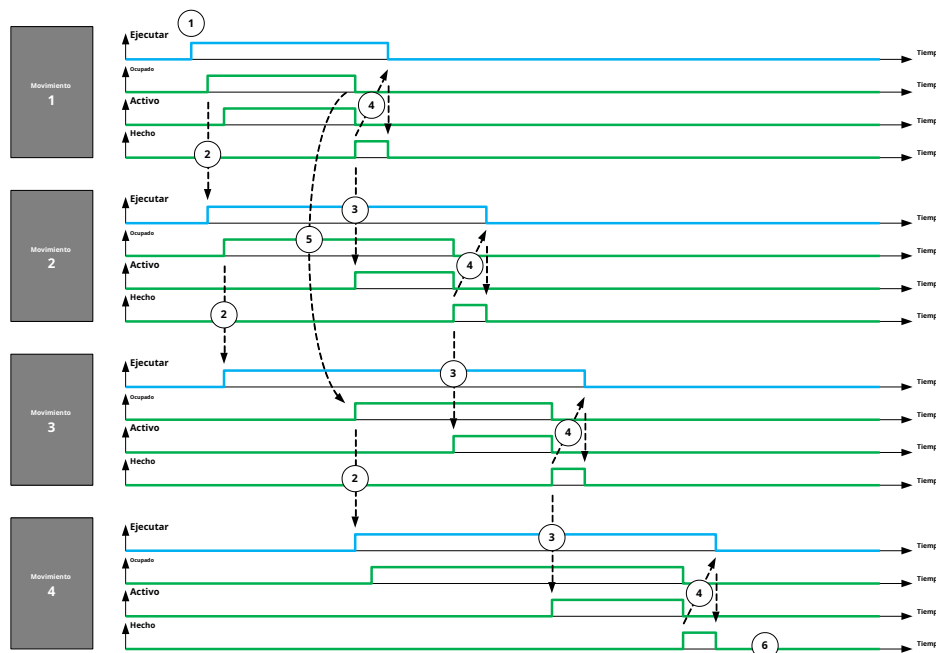


Tabla 3-3 Explicación de la secuencia de señales

No.	Explicación
1	Un flanco ascendente en la entrada Ejecutar de la primera función de movimiento inicia la secuencia de movimiento.
2	Una vez que la función de movimiento indica Ocupado, la siguiente función de movimiento puede iniciarse con un flanco ascendente en la entrada Ejecutar.
3	Debido al almacenamiento en búfer de la función de movimiento en el controlador del robot, el movimiento se activa cuando se completa la función de movimiento anterior.
4	Si la función de movimiento indica Hecho, el movimiento se ha completado y la entrada Ejecutar se puede restablecer.
5	Si la memoria intermedia de trabajos en el controlador del robot no puede aceptar más trabajos, la señal de ocupado se retrasará hasta que se puedan transferir nuevos trabajos al controlador del robot.
6	Con la señal Done de la última función de movimiento de la secuencia, el movimiento La secuencia ha sido completamente procesada.

4 Generación de programas

El objetivo de este capítulo es ofrecerle una explicación básica de cómo crear un programa de usuario para controlar un robot KUKA utilizando la biblioteca KUKA.PLC mxAutomation.

Nota

Para obtener información más detallada sobre la función y los parámetros de los bloques de la biblioteca KUKA.PLC mxAutomation, consulte la documentación de KUKA correspondiente.

4.1 Conexión del robot KUKA al SIMATIC S7

4.1.1 Robot como dispositivo PROFINET IO

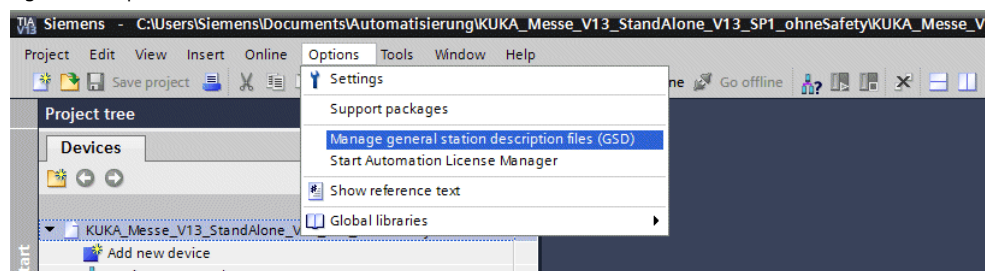
El robot KUKA está integrado en la configuración de hardware del proyecto TIA Portal como dispositivo PROFINET IO. El robot se integra a través de un archivo GSDML para integrarse adicionalmente en el TIA Portal que contiene la descripción del hardware del robot y los posibles telegramas de datos para el intercambio de datos entre el controlador SIMATIC y el robot.

4.1.2 Instalación del archivo GSDML del robot

El archivo GSDML del robot generalmente se suministra con el robot o la documentación del robot o puede obtenerse del departamento de servicio de KUKA.

El archivo GSDML se puede integrar en el TIA Portal mediante la función de gestión de archivos GSD.

Figura 4-1 Importación del archivo GSDML



4.1.3 Integración del robot en la configuración del hardware

Cuando el archivo GSDML se ha integrado en el TIA Portal, el controlador de robot KUKA KR C4 con la versión de firmware adecuada se puede integrar en la configuración de hardware del TIA Portal mediante la función de arrastrar y soltar.

Al hacer doble clic en el controlador del robot en la configuración de hardware, puede configurar los componentes para el intercambio de datos con el controlador SIMATIC.

Para poder intercambiar los datos necesarios para controlar el robot utilizando el controlador SIMATIC, configure el telegrama de intercambio de datos requerido con el controlador del robot en la configuración de hardware. Para ello, utilice arrastrar y soltar para integrar al menos el telegrama para 256 entradas y salidas digitales de la selección del telegrama en el componente del controlador de robot de la configuración de hardware.

4.1 Conexión del robot KUKA al SIMATIC S7

Figura 4-2 Selección del robot KUKA del catálogo de hardware

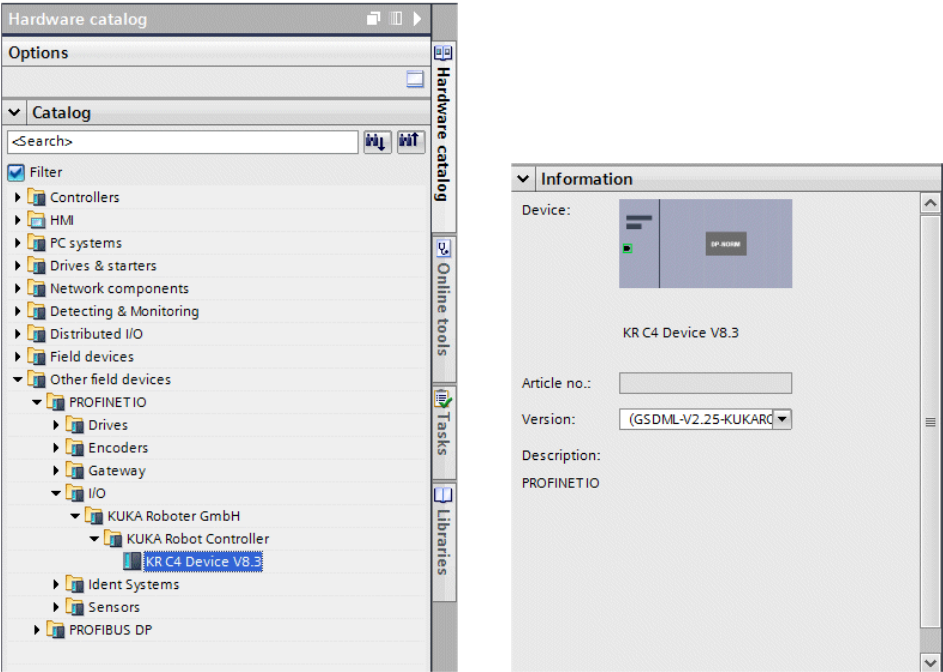


Figura 4-3 Intercambio de datos entre el controlador SIMATIC y el robot

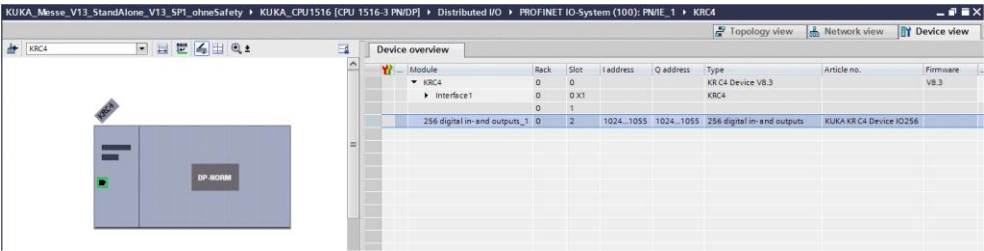


Figura 4-4 Telegrama de datos

Device overview								
	Module	Rack	Slot	I address	Q address	Type	Article no.	Firmware
	▼ KRC4	0	0			KR C4 Device V8.3		V8.3
	▶ Interface 1	0	0 X1			KRC4		
	256 digital in-and outputs_1	0	2	1024...1055	1024...1055	256 digital in- and outputs	KUKA KR C4 Device IO256	

Nota

Opcionalmente, telegramas adicionales o extensiones de telegramas para funciones opcionales como Safety Integrated se pueden integrar en el controlador del robot en este punto. En t.

En la configuración de hardware del TIA Portal, utilice la función de arrastrar y soltar para integrar el telegramas adicionales en el controlador del robot.

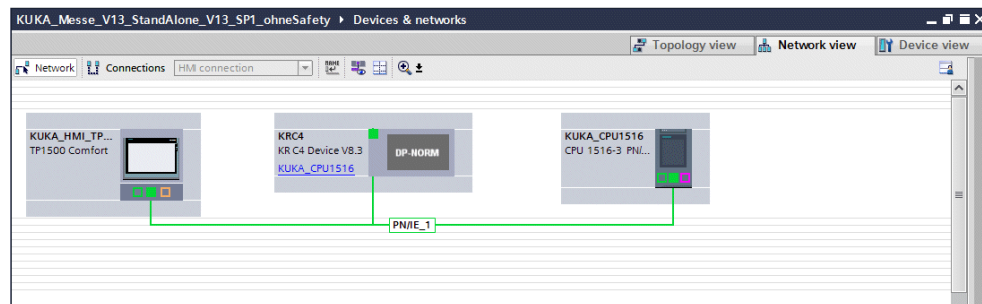
4.1 Conexión del robot KUKA al SIMATIC S7

4.1.4 Conexión del SIMATIC S7 al robot

Por último, utilice una conexión PROFINET para conectar el controlador del robot al controlador SIMATIC. Para ello, asigne direcciones IP adecuadas en los dispositivos individuales o en la configuración del hardware.

Opcionalmente, un dispositivo HMI utilizado para controlar y supervisar las funciones del robot se puede integrar en la configuración en este punto.

Figura 4-5 Conexión del robot, controlador SIMATIC y HMI



4.1.5 Control de varios robots con un SIMATIC S7

Con un controlador SIMATIC adecuado, se pueden controlar hasta cinco robots KUKA por separado en la configuración estándar con la ayuda de la biblioteca de bloques KUKA.PLC mxAutomation.

Para ello, el bloque de datos suministrado con la librería de bloques KUKA.PLC mxAutomation ofrece cinco conjuntos de datos para el intercambio de datos entre el controlador SIMATIC y el robot.

La asignación de los datos al robot correspondiente se realiza a través de las direcciones de telegrama definidas en el rango de E / S del controlador SIMATIC como se define en la configuración de hardware.

La asignación de los bloques de función o funciones del robot a los robots individuales en el programa de usuario se realiza a través de la entrada AxisGroupIdx de los bloques que representan el índice de la matriz de datos en el bloque de datos MxADBRobots.

Sin embargo, esta funcionalidad se explicará en detalle en los siguientes capítulos.

Figura 4-6 Grupo de datos para 5 robots en el bloque de datos MxADBRobots

MxADBRobots				
	Name	Data type	Offset	Start value
1	Static			
2	KRC_AxisGroupRefArr	Array[1..5] of *AXIS_GROUP_REF*	0.0	
3	KRC_AxisGroupRefArr[1]	*AXIS_GROUP_REF*	0.0	
4	KRC_AxisGroupRefArr[2]	*AXIS_GROUP_REF*	736.0	
5	KRC_AxisGroupRefArr[3]	*AXIS_GROUP_REF*	1472.0	
6	KRC_AxisGroupRefArr[4]	*AXIS_GROUP_REF*	2208.0	
7	KRC_AxisGroupRefArr[5]	*AXIS_GROUP_REF*	2944.0	
8	PLC_SYS_TICK_CNT	DInt	3680.0	0
9	PLC_TICKS_PER_SEC	Int	3684.0	0

4.2 Importando la biblioteca de bloques

4.2.1 Extrayendo la biblioteca de bloques

Antes de poder utilizar la biblioteca de bloques KUKA.PLC mxAutomation en su programa de usuario, debe aplicar los bloques de función de la biblioteca de bloques a su proyecto TIA Portal.

La biblioteca de bloques KUKA.PLC mxAutomation para TIA Portal se suministra como una biblioteca global desde la cual todos los bloques de función de la biblioteca de bloques se pueden aplicar a un proyecto de TIA Portal mediante la función de arrastrar y soltar.

Para poder utilizar la biblioteca global para TIA Portal, extraiga el archivo de la biblioteca global a una ubicación adecuada en el disco duro de su programador.

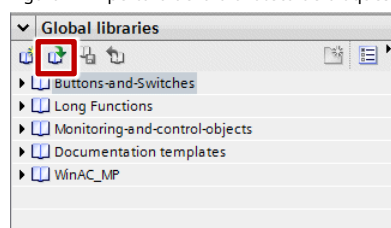
Nota

La biblioteca de bloques KUKA.PLC mxAutomation para TIA Portal se puede solicitar al servicio técnico de KUKA Roboter GmbH o a la línea directa de la empresa.

4.2.2 Abrir la biblioteca de bloques

Después de extraer, puede utilizar las funciones de acceso del TIA Portal para abrir la biblioteca global.

Figura 4-7 Apertura de la biblioteca de bloques



La biblioteca de bloques KUKA.PLC mxAutomation se almacena en la biblioteca global como copia maestra.

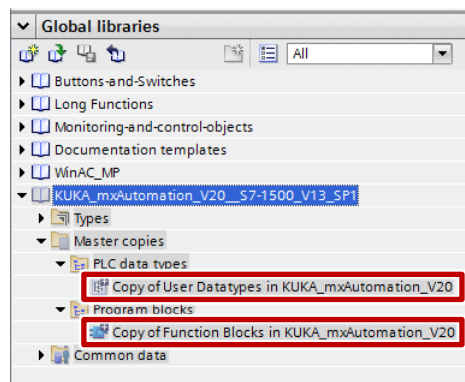
4.2.3 Transferencia de bloques al programa de usuario

Simplemente use arrastrar y soltar para transferir la biblioteca de bloques KUKA.PLC mxAutomation desde la biblioteca global a su programa de usuario. Transfiera los tipos de datos de PLC necesarios y los bloques de función a su proyecto de TIA Portal.

Primero arrastre el objeto incluido en las copias maestras de la biblioteca global en tipos de datos de PLC a la carpeta de tipos de datos de PLC en su proyecto. En esta carpeta, se crea automáticamente una subcarpeta que contiene todos los tipos de datos de PLC de la biblioteca de bloques KUKA.PLC mxAutomation.

A continuación, arrastre el objeto incluido en Bloques de programa a la carpeta Bloques de programa de su proyecto TIA Portal. También en esta carpeta se crea automáticamente una subcarpeta que contiene todos los bloques de función de KUKA.PLC mxAutomation biblioteca de bloques.

Figura 4-8 Transferencia de bloques y tipos de datos del PLC al programa de usuario



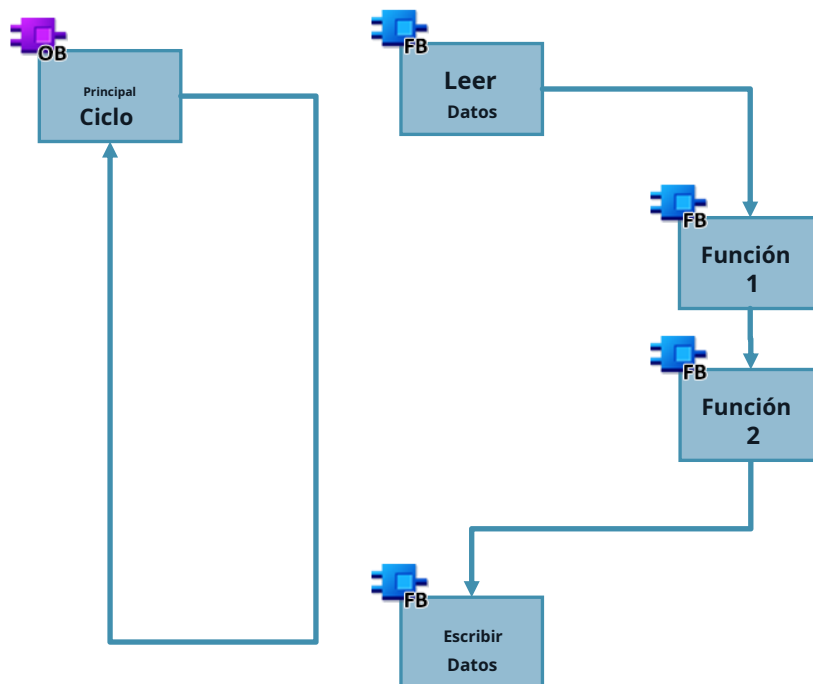
Ahora su proyecto de TIA Portal incluye todos los tipos de datos y bloques de función necesarios de la biblioteca de bloques KUKA.PLC mxAutomation. Ahora puede utilizar las funciones de la biblioteca de bloques en su programa de usuario.

4.3 Estructura básica del programa

4.3.1 Concepto

Conceptualmente, el intercambio de datos entre el controlador SIMATIC y el robot KUKA con la ayuda de la biblioteca de bloques KUKA.PLC mxAutomation sigue la funcionalidad del controlador SIMATIC. Sin embargo, este concepto debe implementarse en el programa de usuario del controlador SIMATIC utilizando los bloques de biblioteca correspondientes.

Figura 4-9 Concepto de intercambio de datos entre SIMATIC y robot



4.3 Estructura básica del programa

En el bloque de organización donde se va a procesar el programa del robot, esto requiere que la función de lectura de datos del robot desde el controlador del robot se llame como el primer bloque. Como última función en el bloque de organización, coloque los datos calculados de escritura en la función del controlador del robot. Entre estos dos bloques, puede colocar los bloques de función para implementar el programa de robot real en el controlador SIMATIC.

Para la llamada del programa de robot en el controlador SIMATIC, generalmente es suficiente utilizar el bloque de organización estándar (ciclo principal u OB 1).

4.3.2 Estructura del programa

Para estructurar un programa de usuario para el control de un robot industrial KUKA utilizando la biblioteca de bloques KUKA.PLC mxAutomation, podemos ofrecer la siguiente recomendación:

Para cada robot que se va a controlar mediante el controlador SIMATIC, se debe utilizar un bloque de función independiente que se llama en el OB de ciclo de programa (OB 1).

Dentro del bloque de funciones de un robot, el bloque para leer los datos del robot debe llamarse en la primera red y el bloque para escribir los datos del robot debe llamarse en la última red.

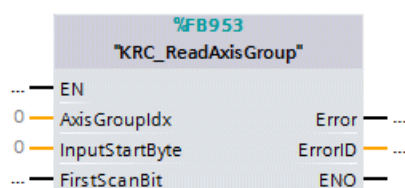
Las llamadas, entre las llamadas anteriores, de los otros bloques de función de la biblioteca de bloques KUKA.PLC mxAutomation también se pueden dividir en otros bloques de función estructurados por función.

Todas las llamadas de bloques de la biblioteca KUKA.PLC mxAutomation deben implementarse como llamadas de instancias múltiples en los bloques de llamada.

4.3.3 Leer datos del robot

El bloque KRC_ReadAxisGroup (FB 953) de la biblioteca KUKA.PLC mxAutomation le permite leer los datos del controlador del robot en la gestión de datos interna de la biblioteca de bloques KUKA.PLC mxAutomation.

Figura 4-10 Leer datos del robot



Esto proporciona los datos del robot o controlador de robot al programa de usuario en el controlador SIMATIC para los otros bloques de la biblioteca KUKA.PLC mxAutomation.

La asignación de los datos al robot correspondiente conectado al controlador SIMATIC se realiza a través de la entrada AxisGroupIdx que corresponde al índice de la matriz de datos en el bloque de datos MxADBRobots y la entrada InputStartByte que informa al bloque de la dirección de telegrama del robot desde el IO gama del controlador SIMATIC.

Entonces, solo el índice de la entrada AxisGroupIdx se puede utilizar para asignar el robot a los bloques de programa del programa del robot en SIMATIC.

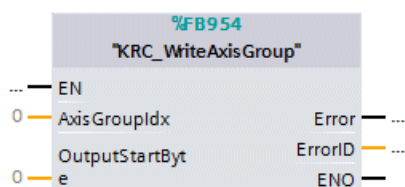
4.3.4 Datos de proceso en SIMATIC S7

Después de leer los datos del controlador del robot, todos los bloques de función de la biblioteca KUKA.PLC mxAutomation se pueden utilizar y llamar. Los bloques utilizan los datos leídos almacenados en el bloque de datos MxADBRobots.

4.3.5 Escribir datos en el robot

Después de procesar el programa de robot en el controlador SIMATIC, el bloque KRC_WriteAxisGroup (FB 954) de la biblioteca KUKA.PLC mxAutomation transfiere los datos del bloque de datos MxADBRobots al robot definido por la dirección del telegrama.

Figura 4-11 Escribir datos en el robot



AVISO

Al ingresar las direcciones de telegramas del robot para lectura / escritura, asegúrese de que la asignación sea correcta. De lo contrario, puede ocurrir un comportamiento inesperado del robot.

4.4 Programación de funciones básicas

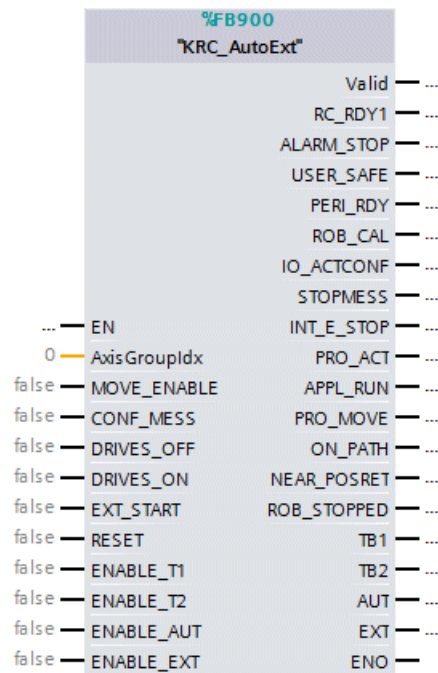
4.4.1 Encender el robot

El robot se enciende mediante el bloque KRC_AutoExt. Esto requiere que las entradas del bloque se operen en una secuencia de señal específica y dependiendo de las salidas del bloque.

Nota

Para poder operar el robot utilizando la biblioteca de bloques KUKA.PLC mxAutomation, el controlador del robot debe estar en modo EXT. Puede que sea necesario seleccionar primero este modo utilizando el interruptor de llave del KUKA SmartPad.

Figura 4-12 Encendido del robot mediante el bloque de funciones KRC_AutoExt



El encendido del robot mediante el bloque KRC_AutoExt se realiza mediante la siguiente secuencia de señales.

Figura 4-13 Secuencia de señales estándar para encender el robot

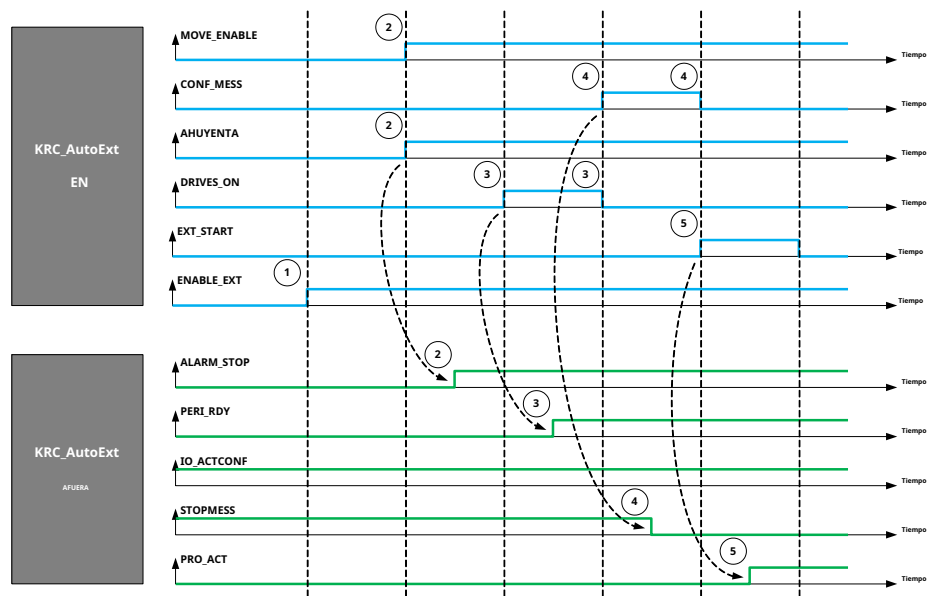


Tabla 4-1 Explicación de la secuencia de señales

No.	Explicación
1	Para poder operar el robot utilizando la biblioteca de bloques KUKA.PLC mxAutomation, primero debe seleccionarse el modo EXT en el bloque KRC_AutoExt. Esto se logra configurando la señal ENABLE_EXT en el bloque de funciones. Nota: El cambio al modo EXT solo es posible si este modo está seleccionado en el robot.
2	Luego configure las señales de entrada MOVE_ENABLE y DRIVES_OFF del bloque. El robot responde a estas señales configurando la señal de salida ALARM_STOP.
3	Luego configure la señal de entrada DRIVES_ON. El robot responde indicando que la periferia está lista a través de la señal de salida PERI_RDY. Una vez que la periferia está lista, puede restablecer las señales de entrada DRIVES_ON en el bloque.
4	Luego, reconozca cualquier mensaje pendiente en el controlador del robot configurando la entrada CONF_MESS. Después de reconocer los mensajes, el robot establece la salida STOPMESS del bloque como una señal de retroalimentación. La entrada CONF_MESS se puede restablecer después de la señal de retroalimentación del robot.
5	Luego, encienda el robot configurando la señal EXT_START. El controlador del robot indica que está listo para funcionar y el robot se enciende correctamente configurando la señal de salida PRO_ACT. Ahora el robot está encendido y puede manejarse con los bloques de función de la biblioteca de bloques KUKA.PLC mxAutomation.

Nota

Un bloque preparado de SIEMENS Robotics Support está disponible para encender el robot. Puede obtenerse de la dirección que figura en el Apéndice.

4.4.2 Inicializando el robot

Antes de que se puedan intercambiar datos entre SIMATIC S7 y el robot, se debe inicializar la interfaz KUKA.PLC mxAutomation. Se inicializa llamando al bloque KRC_Initialize.

La inicialización exitosa de la interfaz se devuelve al programa de usuario a través de la salida Done.

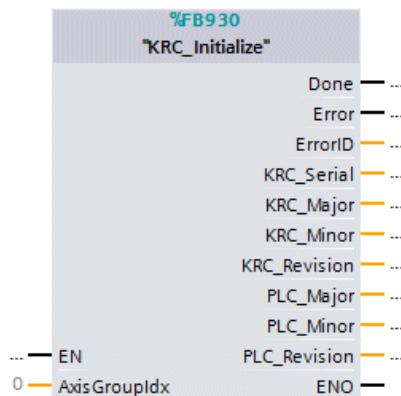
Además, el bloque devuelve los ID de versión del intérprete KUKA.PLC mxAutomation y la biblioteca de bloques KUKA.PLC mxAutomation y el número de serie de el controlador del robot:

Número de serie del controlador de robot: KRC_Serial

Versión del intérprete: KRC_Major. KRC_Minor. KRC_Revision Versión

de la biblioteca de bloques: PLC_Major. PLC_Minor. PLC_Revision

Figura 4-14 Inicialización del robot



4.4.3 Comprobación del estado del robot

El estado del robot o el estado actual del robot debe determinarse mediante varios bloques de función de la biblioteca de bloques KUKA.PLC mxAutomation.

Comprobación del estado del robot

El bloque de funciones **KRC_ReadMXAStatus** le permite determinar el estado actual de la interfaz de datos de la biblioteca KUKA.PLC mxAutomation.

Figura 4-15 Comprobación del estado del robot



A través del tipo de datos del PLC **MXA_STATUS**, la información de estado se devuelve en una salida del bloque que puede tener los siguientes estados:

Los bloques de función de la biblioteca KUKA.PLC mxAutomation no pueden intercambiar datos con el controlador del robot.

Hay un error activo en la interfaz de datos.

El intérprete del controlador del robot no está activo, en espera o procesando un programa actualmente.

El robot se detuvo y todas las instrucciones se cancelaron.

Leer mensajes de error del controlador del robot

El bloque de funciones **KRC_ReadKRCError** le permite determinar el error actual estado del controlador del robot.

Figura 4-16 Lectura de mensajes de error del controlador del robot



El controlador del robot contiene un búfer de mensajes para varios mensajes de error. Con una llamada del bloque de funciones **KRC_ReadKRCErrors**, se pueden leer simultáneamente hasta diez mensajes del búfer de mensajes. Además, el bloque contiene una entrada de compensación que le permite leer los mensajes de error restantes en grupos de diez mensajes.

Nota

Los mensajes de error del controlador del robot deben confirmarse a través de la entrada **CONF_MESS** del bloque de función **KRC_AutoExt** de la biblioteca de bloques **KUKA.PLC mxAutomation**.

Escaneo de señales de diagnóstico del controlador del robot

El bloque de función **KRC_Diag** le permite determinar varios datos de diagnóstico del controlador del robot o enviar un intérprete para procesar los comandos de los bloques de función desde la biblioteca de bloques **KUKA.PLC mxAutomation** en el controlador del robot.

El bloque de funciones del controlador **SIMATIC** proporciona los siguientes datos de diagnóstico:

Número de instrucciones almacenadas en búfer de los bloques de función de la biblioteca de bloques **KUKA.PLC mxAutomation** en el controlador del robot.

Señal de latido del intérprete de envío que procesa los comandos de los bloques de función de la biblioteca **KUKA.PLC mxAutomation** en el controlador del robot como monitorización de latido.

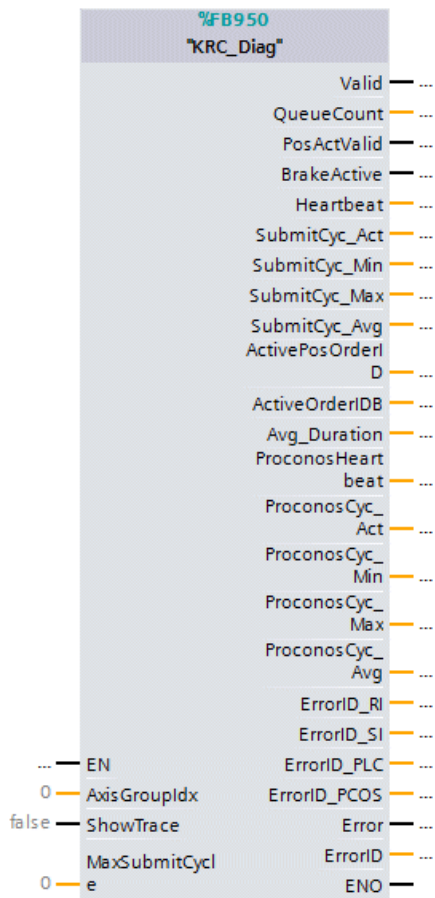
Tiempo de ciclo actual, más corto, más largo y promedio del intérprete enviado.

Número de error del intérprete del robot.

Número de error del intérprete de envío.

Número de error del PLC en el controlador del robot.

Figura 4-17 Escaneo de señales de diagnóstico del controlador del robot

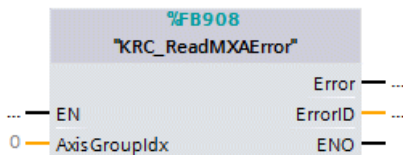


Nota Para cada robot, este bloque de funciones solo se puede llamar o instanciar una vez. Si el bloque se llama varias veces, pueden producirse incoherencias de datos al emitir datos.

Lectura de mensajes de error de la interfaz de automatización KUKA.PLC mx

El bloque de funciones KRC_ReadMXAError le permite determinar el estado de error actual de un grupo de ejes. La causa del error se emite como un número de error a través del ErrorID producción.

Figura 4-18 Lectura de mensajes de error de la interfaz KUKA.PLC mxAutomation



4.4.4 Reconocer mensajes de error

Para reconocer los distintos mensajes de error, la biblioteca de bloques KUKA.PLC mxAutomation ofrece varios bloques funcionales.

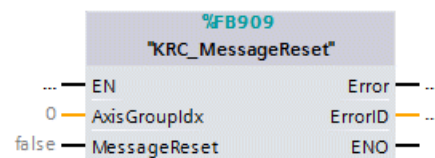
Nota

Los mensajes de error del controlador del robot solo se pueden reconocer cuando el robot está parado.

Confirmación de mensajes de error de la interfaz KUKA.PLC mxAutomation

El bloque de funciones KRC_MessageReset permite reconocer el estado de error actual de un grupo de ejes de la interfaz KUKA.PLC mxAutomation.

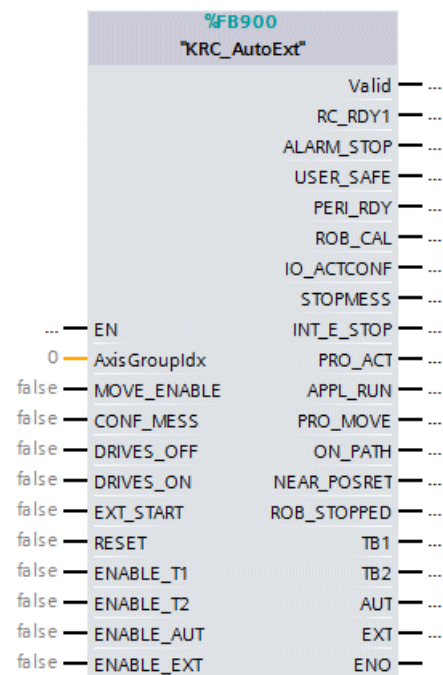
Figura 4-19 Confirmación de mensajes de error de la interfaz KUKA.PLC mxAutomation



Reconocimiento de mensajes de error del controlador del robot

Usando el bloque de función KRC_AutoExt, mensajes de error del robot. El controlador puede ser reconocido por un flanco ascendente en la entrada CONF_MESS.

Figura 4-20 Confirmación de mensajes de error del controlador del robot



Nota

Este bloque también se utiliza para encender las unidades de potencia del robot y seleccionar los modos del controlador del robot.

4.4.5 Leer la posición del robot

La biblioteca de bloques KUKA.PLC mxAutomation proporciona dos bloques de funciones diferentes para leer la posición actual del robot.

Leer las posiciones de los ejes del robot

El bloque KRC_ReadActAxisPos se utiliza para leer cíclicamente la posición actual de los ejes del robot.

Figura 4-21 Posiciones de los ejes del robot

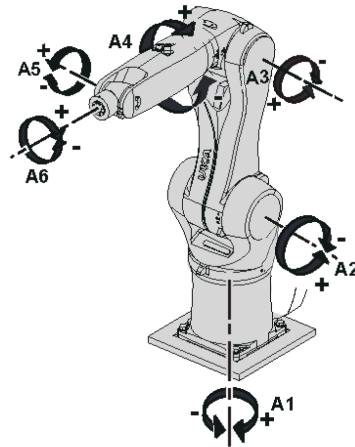
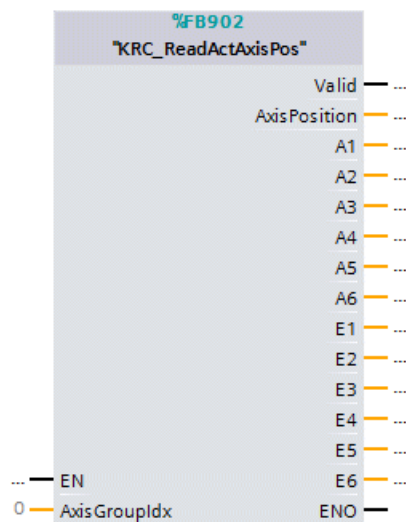


Figura 4-22 Lectura de las posiciones de los ejes del robot



4.4 Programación de funciones básicas

La salida AxisPosition se utiliza para generar la posición de los ejes del robot en la estructura E6AXIS definida. Además, las salidas A1 a A6 contienen la misma información que los valores individuales.

Las salidas E1 a E6 muestran la posición actual de los ejes externos que también se pueden influir mediante el controlador del robot.

Leer la posición del robot en el sistema de coordenadas cartesianas

El bloque KRC_ReadActualPosition se utiliza para leer cíclicamente la posición actual del robot o el punto central de la herramienta (TCP) en el sistema de coordenadas cartesianas seleccionado actualmente.

Figura 4-23 Posiciones del robot cartesiano

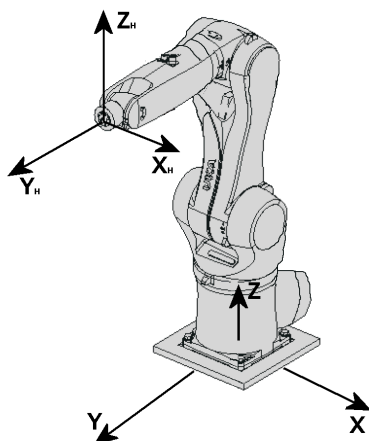
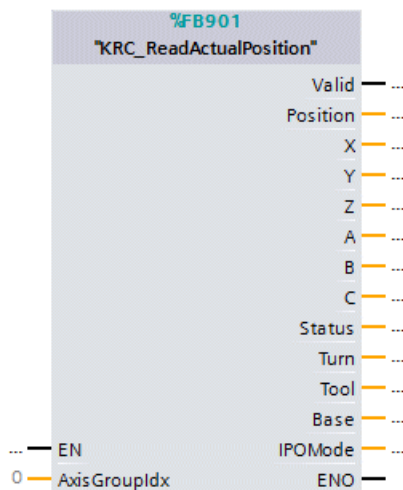


Figura 4-24 Lectura de la posición de la herramienta en el sistema de coordenadas cartesianas



La salida de posición se utiliza para dar salida a la posición actual en la estructura 6POS definida. Además, las salidas X a Z y A a C contienen la misma información que los valores individuales para la posición en el espacio y la rotación sobre los ejes de coordenadas. Los valores de Estado y Giro se utilizan para especificar la orientación del robot en la posición con mayor detalle.

A través de las salidas de Herramienta y Base, se emiten adicionalmente la herramienta actualmente activa y el número del sistema de coordenadas seleccionado actualmente.

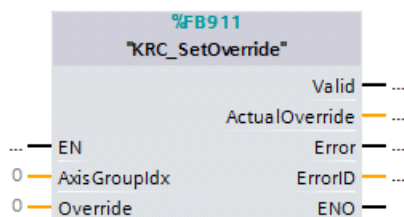
Nota

Para obtener información relacionada con los sistemas de coordenadas del robot, consulte el Apéndice de este documento.

4.4.6**Configuración de la anulación de velocidad**

La función de anulación es la función más importante al iniciar y mover manualmente el robot. Esta función le permite influir en la velocidad especificada del robot entre 0% y 100%.

Figura 4-25 Bloque KRC_SetOverride



Para poder utilizar la función de anulación en cualquier momento, se recomienda llamar al bloque de función KRC_SetOverride en cada ciclo del PLC y transferir la configuración de anulación actual al robot.

Nota

Siempre inicie movimientos de robot recién programados o movimientos de desplazamiento manual con una anulación de velocidad reducida y aumente la velocidad solo cuando el robot ejecute el movimiento deseado y esperado.

4.5**Programación de funciones de movimiento****4.5.1****Tipos de movimiento**

Hay dos opciones para mover un robot:

Movimiento específico del eje

Los ejes individuales del robot se mueven de forma completamente independiente entre sí. No hay interpolación entre los ejes. Por lo tanto, la trayectoria del robot es difícil de predecir y depende de la posición actual de los ejes del robot. El movimiento se ejecuta a la velocidad máxima programada, en porcentaje, de los ejes individuales o limitado por esta velocidad.

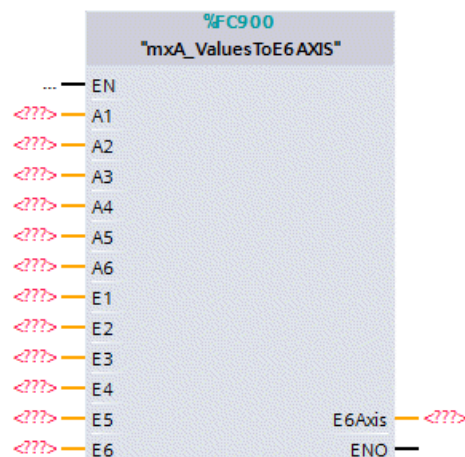
Movimiento cartesiano

Los ejes individuales del robot se mueven coordinados entre sí a través de la transformación de coordenadas para que el punto de referencia del robot se mueva a lo largo de la ruta especificada. De esta manera, el punto de referencia del robot se puede mover en línea recta o en una trayectoria circular con coordenadas especificadas en el sistema de coordenadas seleccionado. El movimiento punto a punto en el sistema de coordenadas también es posible.

4.5.2 Definición de coordenadas

El movimiento específico del eje requiere que la posición final de los ejes individuales se especifique mediante el tipo de datos del PLC E6AXIS. Para definir los valores de los ejes individuales en este tipo de datos, la biblioteca de bloques KUKA.PLC mxAutomation proporciona una función que cambia las posiciones individuales de los ejes al tipo de datos del PLC E6AXIS.

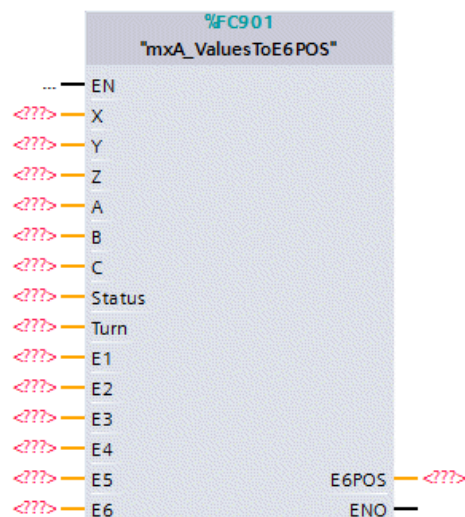
Figura 4-26 Función para configurar el tipo de datos del PLC E6AXIS



Asimismo, para el movimiento cartesiano, la biblioteca KUKA.PLC mxAutomation proporciona una función para definir las coordenadas X, Y y Z y rotar alrededor de los ejes de coordenadas A, B y C en el tipo de datos del PLC E6POS.

Además, los parámetros de estado y giro para la definición precisa de la posición del eje se puede incluir en este tipo de datos.

Figura 4-27 Función para configurar el tipo de datos del PLC E6POS

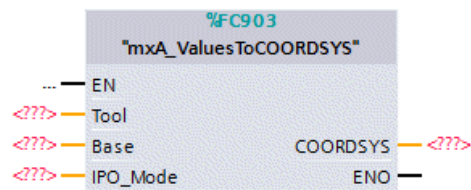


Nota

Las dos funciones para configurar los tipos de datos del PLC, E6AXIS y E6POS, incluyen adicionalmente parámetros de entrada para seis ejes externos, E1 a E6, que se pueden utilizar para controlar ejes conectados adicionalmente al controlador del robot como transportadores, ejes de herramientas, etc.

El movimiento cartesiano requiere que, además de la posición final del movimiento, se seleccione el sistema de coordenadas. Para ello, la librería de bloques KUKA.PLC mxAutomation proporciona una función de entrada para el tipo de datos del PLC COORDSYS.

Figura 4-28 Función para configurar el tipo de datos del PLC COORDSYS



Con esta función, se pueden realizar los siguientes ajustes para todas las funciones de movimiento cartesiano:

Tabla 4-2 Parámetros de la función mxA_ValuesToCOORDSYS

Interfaz	Función
Entradas	
Herramienta	Selecciona el número del sistema de coordenadas de la herramienta o herramienta activa.
Base	Selecciona el número del sistema de coordenadas base que se utilizará.
IPO_Mode	Selecciona el modo de interpolación, es decir, si el sistema de coordenadas de la herramienta de coordenadas de la herramienta o el sistema de coordenadas de la base se utilizará como base para el movimiento.
Salidas	
COORDSYS	Estructura de datos para la definición del sistema de coordenadas de movimientos de desplazamiento cartesianos.

4.5.3 Movimiento de punto a punto (PTP)

En un movimiento de punto a punto, el punto de referencia del robot se mueve al punto final especificado. El controlador del robot determina de forma independiente la trayectoria entre la posición actual del robot y el punto final, de modo que se logre la mayor velocidad de desplazamiento posible del robot durante el movimiento.

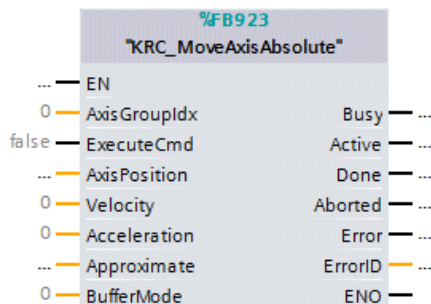
El controlador del robot selecciona la trayectoria de modo que los ejes del robot lentos, como los ejes A4 a A6 preferiblemente ejecutan movimientos cortos mientras que los ejes rápidos del robot se mueven más tiempo distancias.

Con la biblioteca de bloques KUKA.PLC mxAutomation, se pueden ejecutar movimientos de punto a punto con los siguientes comandos:

KRC_MoveAxisAbsolute:

Movimiento de desplazamiento específico del eje de los ejes individuales especificando las posiciones absolutas del eje del punto final.

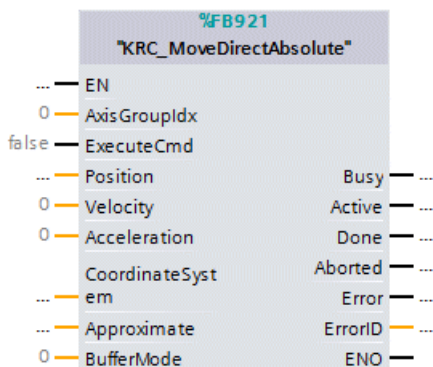
Figura 4-29 Bloque de función KRC_MoveAxisAbsolute



KRC_MoveDirectAbsolute o KRC_MoveDirectRelative:

Movimiento de desplazamiento cartesiano especificando la posición absoluta del punto final en el sistema de coordenadas seleccionado o debido a la distancia relativa entre el punto final y la posición actual del robot en el sistema de coordenadas seleccionado.

Figura 4-30 Bloque de función KRC_MoveDirectAbsolute



Nota

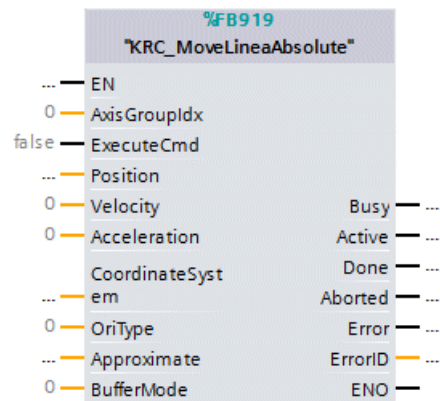
Tenga en cuenta que es difícil o incluso imposible predecir la trayectoria de un movimiento de punto a punto. Especialmente en rangos de movimiento limitados, esto puede resultar en una colisión inesperada del robot con su entorno.

4.5.4

Movimiento lineal

Con la ayuda de los comandos KRC_MoveLinearAbsolute / KRC_MoveLinearRelative, se puede ejecutar un movimiento cartesiano del robot en línea recta entre la posición actual del robot y la posición del punto final especificado como valor absoluto o definido por la distancia. La posición del punto final se especifica en la sistema de coordenadas seleccionado actualmente.

Figura 4-31 Bloque de función KRC_MoveLineaAbsolute



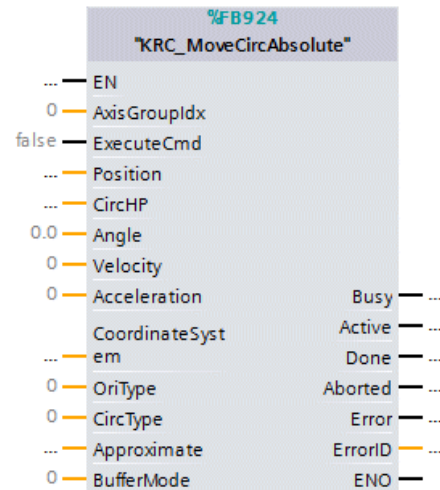
Nota Los movimientos lineales del robot son generalmente más lentos que los movimientos punto a punto, ya que mover el robot en línea recta requiere que todos los ejes del robot se muevan sobre una base definida.

4.5.5 Movimiento circular

Con la ayuda de los comandos KRC_MoveCircAbsolute / KRC_MoveCircRelative, se puede ejecutar un movimiento cartesiano del robot a lo largo de una trayectoria circular entre la posición actual del robot y la posición del punto final especificado como valor absoluto o definido por la distancia.

La definición precisa de la trayectoria circular requiere que se defina un punto auxiliar absoluto adicional. La posición del punto final y el punto auxiliar es especificado en el sistema de coordenadas seleccionado actualmente.

Figura 4-32 Bloque de función KRC_MoveCircAbsolute

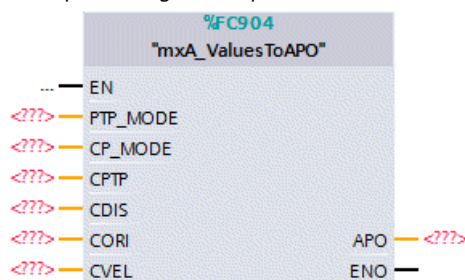


4.5.6 Función de movimiento de posicionamiento aproximado

Los comandos de movimiento enumerados en los capítulos anteriores tienen los llamados parámetros de aproximación que le permiten definir el comportamiento de aproximación del punto final del movimiento. El posicionamiento aproximado permite que el robot ejecute trayectorias compuestas por comandos individuales en un movimiento continuo.

Los parámetros de aproximación para los comandos de movimiento se definen mediante el tipo de datos APO PLC para cuya configuración proporciona una función la biblioteca de bloques KUKA.PLC mxAutomation.

Figura 4-33 Función para configurar el tipo de datos APO PLC



Con esta función, se pueden realizar los siguientes ajustes para todas las funciones de movimiento:

Tabla 4-3 Parámetros de la función mxA_ValuesToAPO

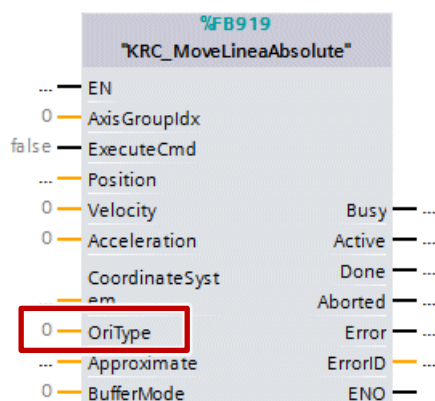
Interfaz	Función
Entradas	
PTP_Mode	Define el movimiento de aproximación para un movimiento de punto a punto.
CP_MODE	Define el movimiento de aproximación para un movimiento interpolado, por ejemplo, el movimiento del robot en línea recta o trayectoria circular.
CPTP	Distancia de aproximación, en porcentaje, para un movimiento de punto a punto en el que, antes del punto final, el posicionamiento aproximado comienza como mínimo. La distancia máxima (100%) representa la mitad de la distancia entre el punto inicial y final.
CDIS	Distancia al punto final, en milímetros, en el que comienza el posicionamiento aproximado como mínimo.
CORI	Parámetro de orientación, en grados, en el que puede comenzar el posicionamiento aproximado, como mínimo, cuando el ángulo de orientación dominante cae por debajo del valor para el punto final establecido aquí.
CVEL	Define la velocidad, en porcentaje, a la que el posicionamiento aproximado en la fase de frenado para el punto final puede comenzar lo antes posible.

Interfaz	Función
Salidas	
APO	Estructura de datos para definir los parámetros de aproximación.

4.5.7 Control de orientación

El control de orientación le permite definir la orientación del punto central de la herramienta (TCP) durante el movimiento.

Figura 4-34 Bloque de función KRC_MoveLineaAbsolute



Para el control de orientación, el parámetro de entrada OriType le permite realizar los siguientes ajustes en los bloques de control de movimiento de la biblioteca KUKA.PLC mxAutomation:

OriType = 0: VAR

La orientación del punto central de la herramienta (TCP) cambia continuamente durante el movimiento.

OriType = 1: CONSTANTE

La orientación del punto central de la herramienta (TCP) permanece constante durante el movimiento.

OriType = 2: JUNTA

La orientación del TCP cambia continuamente durante el movimiento, pero no de manera uniforme. Esto se realiza mediante transformación lineal (movimiento específico del eje) de los ángulos del eje de la muñeca.

4.5.8 Ejes de robot de movimiento manual (JOG)

Para mover manualmente los ejes del robot, la biblioteca de bloques KUKA.PLC mxAutomation proporciona dos bloques funcionales: KRC_JogLinearRelative y KRC_JogToolRelative. Estos dos bloques de función le permiten acercarse a una posición cartesiana definida en el sistema de coordenadas WORLD o el sistema de coordenadas TOOL en línea recta en relación con la posición actual del robot.

Figura 4-35 Bloque de función KRC_JogLinearRelative

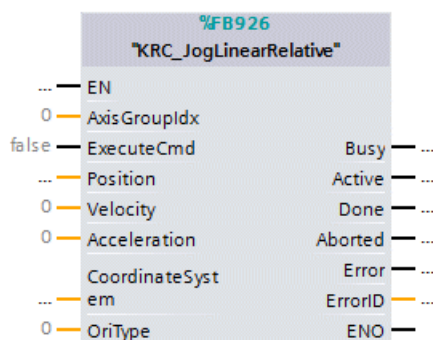
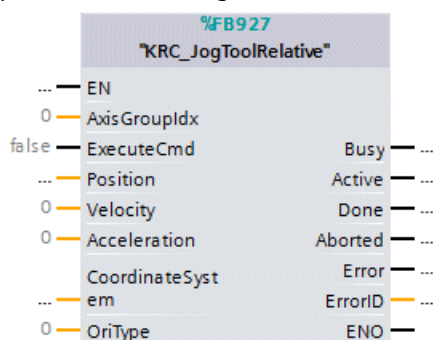


Figura 4-36 Bloque de función KRC_JogToolRelative



El inicio de las funciones jog siempre cancela todas las funciones de movimiento activas o almacenadas en el robot y el robot se desacelera hasta detenerse antes de ejecutar el movimiento jog.

Mover manualmente el eje del robot mediante las teclas de desplazamiento

Si desea mover los ejes del robot mediante las teclas de desplazamiento, la función KRC_MoveLineaAbsolute, para un movimiento en el sistema de coordenadas, o la función KRC_MoveAxisAbsolute, para un movimiento de los ejes del robot, y la función KRC_Abort deben estar interconectadas en un bloque de función. Con las teclas de desplazamiento específicas del eje, se puede iniciar el movimiento del eje respectivo, por ejemplo, con un flanco de señal ascendente de la tecla. Un flanco de señal descendente de la tecla finaliza el movimiento del eje mediante la función KRC_Abort y, por lo tanto, detiene el eje.

Al utilizar este procedimiento, se debe seleccionar un valor de posición poco antes del final de carrera de software del eje para el eje apropiado.

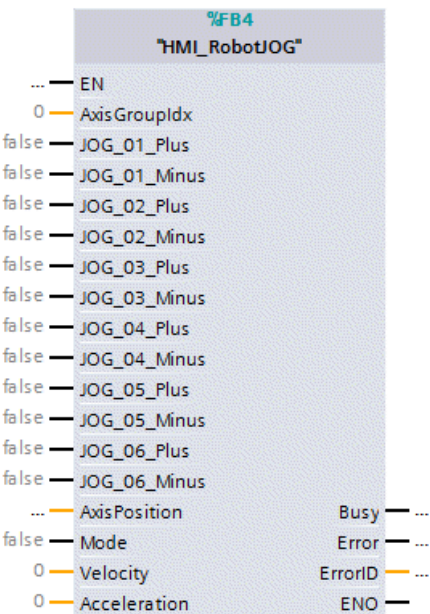
Nota

Un bloque preparado de SIEMENS Robotics Support está disponible para mover el robot usando las teclas de desplazamiento. Puede obtenerse de la dirección que figura en el Apéndice.

La siguiente figura muestra un ejemplo de un bloque de desplazamiento para mover el robot. ejes mediante teclas jog.

4.6 Ejemplo de programa resumido

Figura 4-37 Bloque preparado para mover los ejes del robot con las teclas de desplazamiento



4.6 Ejemplo de programa resumido

El objetivo de este capítulo es ilustrar una vez más la estructura básica de un programa de usuario para controlar un robot industrial KUKA utilizando la biblioteca de bloques KUKA.PLC mxAutomation.

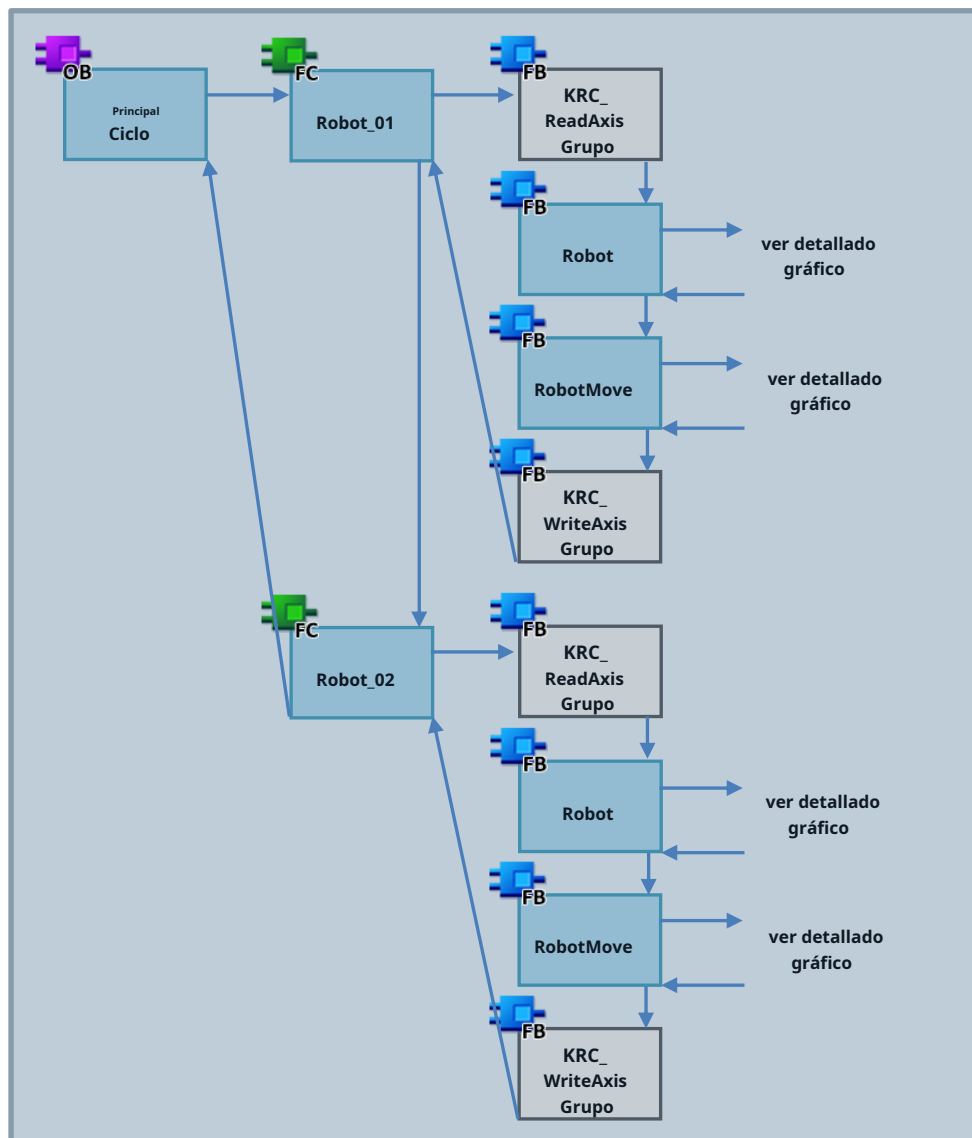
4.6.1 Resumen completo del programa

La siguiente figura muestra un ejemplo de los bloques de función de la librería de bloques y su secuencia de llamada, que son al menos necesarios para trasladar un robot industrial KUKA desde un SIMATIC S7.

El ejemplo de programa está diseñado para controlar dos robots industriales KUKA separados desde el mismo controlador SIMATIC S7.

Para cada robot, la figura muestra los bloques de funciones para leer y escribir los datos del robot. Entre estos bloques, se llaman dos bloques de funciones que contienen el control básico del robot utilizando las funciones administrativas y el control de movimiento del robot. Los dos capítulos siguientes proporcionan una explicación detallada de estos dos bloques.

Figura 4-38 Ejemplo de programa de resumen



4.6.2

Vista detallada del control del robot

El bloque de funciones Robot contiene las funciones administrativas del robot que le permiten ejecutar las siguientes funciones:

Encienda el robot mediante un bloque de funciones

independiente. Inicialice la interfaz KUKA.PLC mxAutomation.

Compruebe el estado de la interfaz de datos KUKA.PLC mxAutomation.

Lea los mensajes de error del controlador del robot, los datos de diagnóstico del robot y los mensajes de error de la interfaz de datos KUKA.PLC mxAutomation.

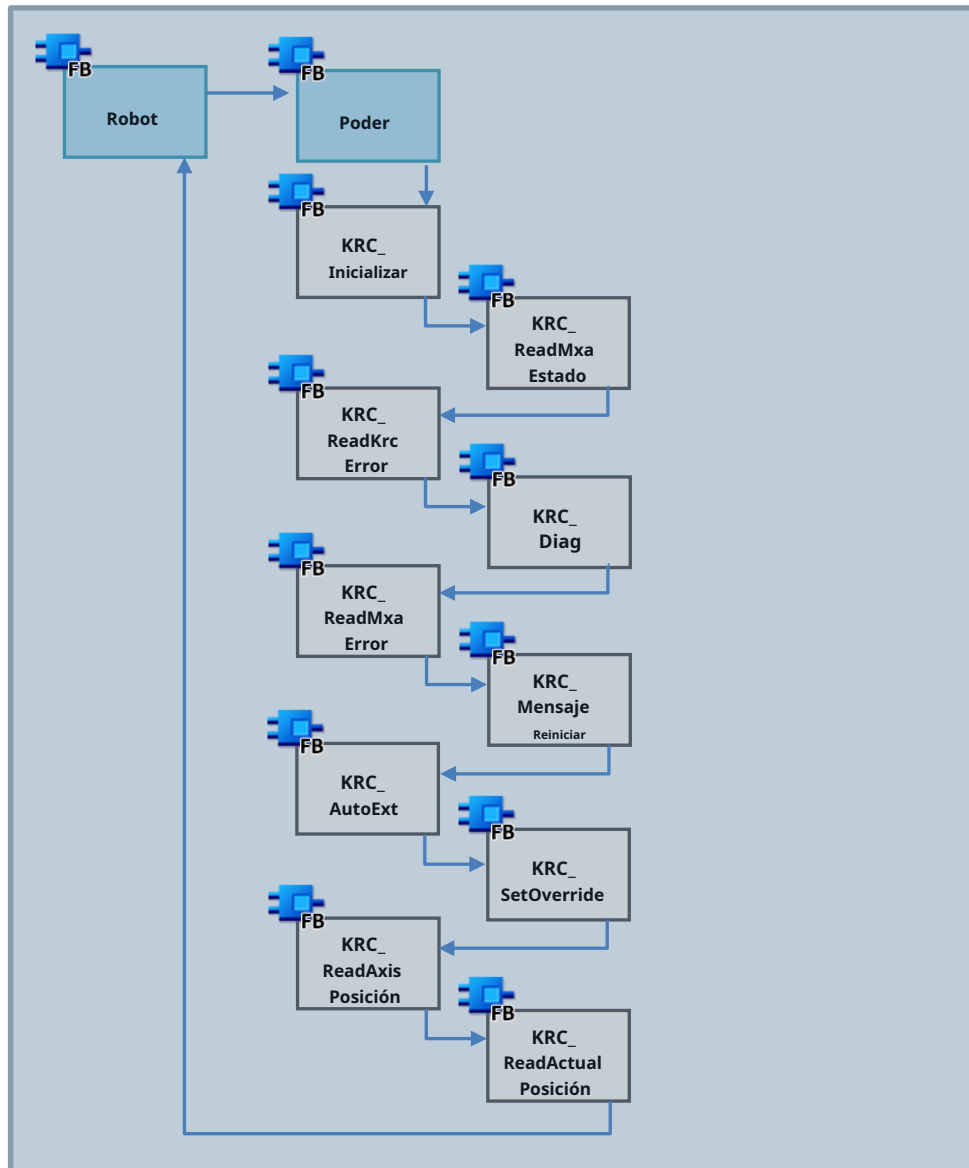
Confirme los mensajes de error en la interfaz de datos KUKA.PLC mxAutomation.

Seleccione el modo de robot / reconozca los mensajes del robot.

Establezca la anulación de los movimientos del robot.

Leer la posición actual del robot en posiciones de eje o coordenadas cartesianas.

Figura 4-39 Vista detallada del bloque de funciones del robot



4.6.3

Vista detallada del control de movimiento

El bloque de funciones RobotMove contiene todos los bloques de funciones para ejecutar los movimientos del robot. Este bloque muestra un ejemplo de la programación de una secuencia de movimiento utilizando diferentes bloques de movimiento de la biblioteca de bloques KUKA.PLC mxAutomation.

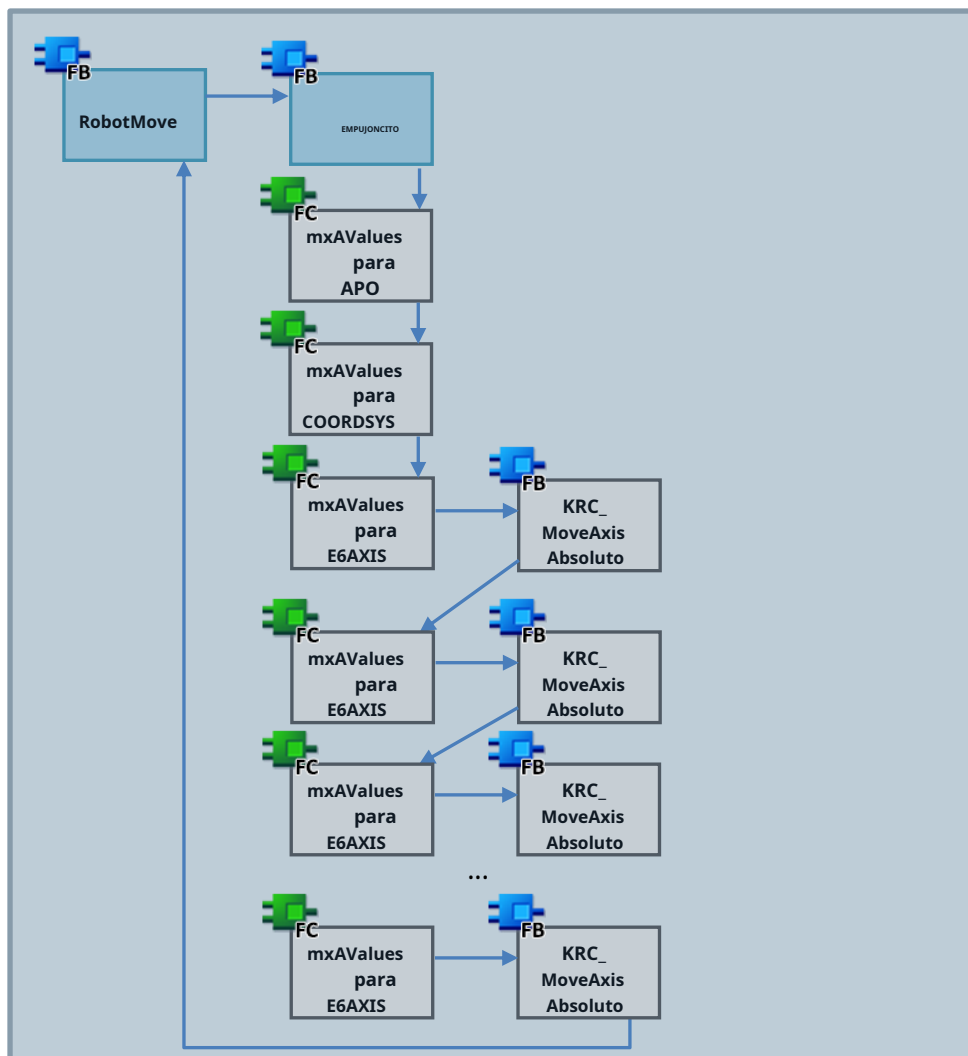
En este bloque, se hace lo siguiente: Primero se establecen los parámetros para cada movimiento usando los parámetros apropiados y luego se llama al bloque de función para el movimiento deseado.

El comienzo del bloque muestra además las funciones para configurar los parámetros para el posicionamiento aproximado de los movimientos y seleccionar la coordenada sistema para ejecutar el movimiento.

4.6 Ejemplo de programa resumido

El bloque preparado para implementar los movimientos de los ejes del robot mediante teclas jog también está integrado al comienzo del bloque de funciones.

Figura 4-40 Vista detallada del bloque de funciones de RobotMove

**Nota**

Si desea que los movimientos individuales sean aproximados, los bloques de función KRC_MoveAxisAbsoluto deben estar interconectados a través de la salida Busy; es decir, una vez que un bloque de funciones indica Ocupado, el siguiente bloque de funciones se puede empezar.

5 Funciones adicionales

El objetivo de este capítulo es presentarle funciones adicionales seleccionadas que van más allá del control básico de un robot industrial KUKA con la ayuda de la biblioteca KUKA.PLC mxAutomation.

5.1 Control de herramientas (pinza)

Para que un robot industrial realice las actividades deseadas, se debe instalar una herramienta en el robot. Estas herramientas generalmente cuentan con numerosas entradas y salidas o un puerto de bus para poder controlar la herramienta como se desee o verificar el estado actual de la herramienta.

Como el robot se controla mediante el controlador SIMATIC S7 con la ayuda de la biblioteca de bloques KUKA.PLC mxAutomation, el control de herramientas tiene dos opciones para transferir los comandos deseados a la pinza.

5.1.1 Control mediante el controlador del robot

El controlador de robot de un robot industrial KUKA también cuenta con un PLC integrado que le permite controlar las entradas y salidas de una herramienta.

Esto le permite conectar las entradas y salidas de una herramienta también directamente al controlador del robot.

Ventajas

Este tipo de control aporta las siguientes ventajas:

El robot y la herramienta forman una unidad de hardware.

La herramienta se puede controlar controlando las entradas y salidas del PLC del controlador del robot con la ayuda de los bloques KRC_ReadDigitalOutput y KRC_WriteDigitalOutput de la biblioteca de bloques KUKA.PLC mxAutomation.

Desventajas

Este tipo de control conlleva las siguientes desventajas:

Los datos de control de la herramienta deben transferirse primero al controlador del robot.

Si es necesario transferir de forma constante varios bits de datos a la herramienta al mismo tiempo, el programa debe garantizar la coherencia de los datos, lo que aumenta el trabajo de programación necesario en el controlador SIMATIC S7 y en el controlador del robot.

Si la herramienta se controla mediante un sistema de bus desde el controlador del robot, la transferencia de datos puede retrasarse debido a la transferencia de datos en el controlador del robot al segundo sistema de bus.

5.1.2 Control directo mediante SIMATIC S7

Dado que el controlador SIMATIC S7 asume el control total y el control de movimiento del robot, la herramienta del robot también se puede aplicar directamente desde SIMATIC S7 o desde E / S distribuidas conectado al controlador.

Si es necesario, la periferia descentralizada del controlador SIMATIC S7 también se puede integrado directamente en la herramienta.

Ventajas

Este tipo de control aporta las siguientes ventajas:

Fácil control de la herramienta desde el programa de usuario del controlador SIMATIC S7 utilizando los comandos de E / S conocidos integrados en el programa para generar los comandos de movimiento del robot.

Opción de integrar las E / S distribuidas del controlador SIMATIC S7 directamente en la herramienta o el brazo robótico, lo que reduce significativamente el cableado de E / S a la herramienta.

Opción de intercambiar datos con las E / S distribuidas mediante comunicación inalámbrica.

Desventajas

Este tipo de control conlleva las siguientes desventajas:

Para activar las funciones de la herramienta es necesario comprobar y comparar el estado actual del robot con el controlador SIMATIC S7.

Es difícil o imposible implementar acciones de herramientas sincronizadas con el movimiento.

5.2 Seguridad integrada

Para utilizar la función Safety Integrated junto con un robot industrial KUKA, KUKA ofrece la opción de seguridad SafeOperation:

5.2.1 Componentes

La opción de seguridad SafeOperation de KUKA incluye los siguientes componentes:

Componentes de software

Tabla 5-1 Componentes de KUKA: opción de seguridad SafeOperation

Componente	No.	Número de artículo	Nota
KUKA.SafeOperation1			Versión: V3.2

Componentes de hardware

Tabla 5-2 Componentes de KUKA: opción de seguridad SafeOperation

Componente	No.	Número de artículo	Nota
Interruptor de referencia módulo	1		Para el control automático de las áreas de seguridad establecidas en el robot.

5.2.2 Funcionalidad

La opción de seguridad SafeOperation de KUKA contiene las siguientes funciones:

Supervisión segura de máx. 16 monitorización cartesiana o específica de eje definida por el usuario espacios.

Supervisión segura de un área de celda definida por el usuario.

Supervisión segura de velocidades específicas del eje.

Monitorización segura de velocidades específicas del espacio.

Monitorización segura de velocidades cartesianas.

5.3 PROFienergía

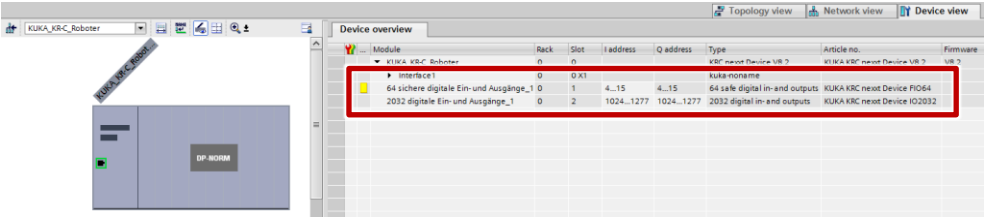
- Modelado de hasta 16 herramientas seguras con un TCP seguro.
- Parada segura usando el sistema de control de seguridad. Parada operativa segura para hasta 6 grupos de ejes. Entradas seguras para activar las funciones de monitorización.
- Salidas seguras para mensajes de estado de las funciones de monitorización. Crear y editar la configuración de seguridad en el controlador del robot.

5.2.3 Control

La opción de seguridad SafeOperation se controla mediante la conexión PROFINET entre el controlador SIMATIC y el controlador del robot con ayuda del telegrama PROFIsafe.

Si desea utilizar esta conexión, simplemente puede integrar la extensión PROFIsafe en la configuración del telegrama del controlador de robot KUKA. Para ello, la extensión del telegrama PROFIsafe se incluye en el archivo GSDML del controlador de robot KUKA.

Figura 5-1 Control de la opción de seguridad SafeOperation a través de PROFIsafe



Nota Como alternativa al intercambio de datos a través de PROFIsafe, las funciones de seguridad del robot también se pueden controlar mediante señales de hardware discretas a través de la interfaz X11 del controlador del robot.

5.3 PROFienergía

En su configuración básica, el robot industrial KUKA ya es un dispositivo PROFienergy que le permite utilizar las opciones de ahorro de energía proporcionadas por PROFienergy. Utilizando telegramas PROFienergy estandarizados, el robot puede configurarse o recuperarse del modo de hibernación.

Con respecto a PROFienergy, el robot industrial KUKA tiene las siguientes posibilidades estados:

Tabla 5-3 Estados PROFienergy del robot industrial KUKA

Estado	Función
Ready_To_Operate	El robot o el controlador de robot está listo para funcionar y puede utilizarse.
Conducir bus APAGADO	Los accionamientos del robot están APAGADOS, el robot no se puede utilizar.

5.4 Recibir mensajes de robot en texto sin formato

Estado	Función
Hibernar	El controlador del robot está en modo Hibernar. No hay comunicación PROFINET. El controlador del robot solo responde al paquete Wake-On-LAN para reactivar el controlador.
Frenos aplicados	El robot o el controlador del robot está listo para funcionar, pero se han aplicado los frenos del robot. El robot responde solo al siguiente comando de movimiento.

Nota

Si la interrupción de la producción es demasiado corta para usar el estado Apagado del bus de impulsión, es posible ahorrar energía usando el estado Frenos aplicados.

Sin embargo, para configurar el robot en este estado, debe estar en parada programada.

Nota

Cuando configure un tiempo de pausa para el robot, es decir, cuando configure el estado de pausa apropiado en el robot, asegúrese de que el tiempo total que seleccione sea mayor que la suma de la fase de quedarse dormido (Time_to_Pause) y la fase de despertar (Time_to_operate) del robot.

Nota

Si se envía un telegrama PROFIenergy al robot, este trabajo se ejecutará directamente en el controlador del robot. Por lo tanto, debe asegurarse de enviar el telegrama PROFIenergy al robot solo cuando esté parado.

5.4 Recibir mensajes de robot en texto sin formato

Además de los bloques para determinar el estado y los mensajes de error del controlador del robot y la interfaz de datos de la biblioteca de bloques KUKA.PLC mxAutomation mencionados en los capítulos anteriores, el usuario tiene la opción adicional de recibir mensajes del robot desde el controlador del robot en texto sin formato. a través de una interfaz UDP.

5.4.1 Principio de funcionamiento

El controlador del robot emite todos los mensajes del robot a través de una conexión UDP en la conexión PROFINET. Estos mensajes pueden recibirse en el controlador SIMATIC y, si es necesario, mostrarse en el panel de operador HMI.

Además, también es posible enviar comandos al controlador del robot a través de la conexión UDP que le permiten configurar el envío de los mensajes del robot. El son posibles los siguientes comandos:

5.4 Recibir mensajes de robot en texto sin formato

Tabla 5-4 Comandos para configurar los mensajes del robot

Mando	Función
SwitchLanguage eng Número de caracteres = 18	Cambie la salida de los mensajes del robot a inglés.
SwitchLanguage deu Número de caracteres = 18	Cambie la salida de los mensajes del robot a alemán.
GetAllKrcMessages AllNoInfo Número de caracteres = 27	Reciba mensajes de texto puros de robot. En este modo de recepción, un área de datos de destino de 150 caracteres es suficiente para el robot. mensajes.

Para transferirlos al controlador del robot, los comandos deben ingresarse en el área de datos de origen para los comandos que se enumeran en la tabla anterior. Para el envío, la tabla también incluye la longitud del comando, es decir, el número de caracteres.

En el controlador SIMATIC, el área de datos de origen para los comandos y el área de datos de destino de los mensajes del robot se crean como una matriz [0..150] de Char.

5.4.2 Programación

La conexión UDP entre el controlador del robot y el controlador SIMATIC se establece mediante el controlador del robot. Por lo tanto, es importante utilizar el ID de conexión especificado por el controlador del robot para los bloques TUSEND y TURCV.

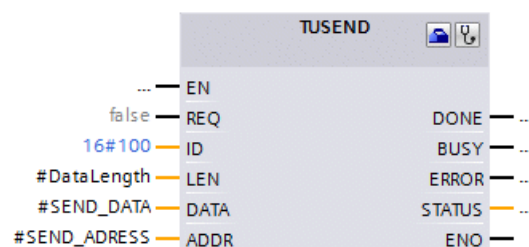
El ID de conexión de 256_{DIC} o 16 # 100_{MALEFICIO} generalmente se configura en el controlador del robot. El ID de conexión debe transferirse a los bloques TUSEND y TURCV a través del Parámetro de ID.

Además, la dirección IP y el puerto de comunicación del controlador del robot deben transferirse a los bloques TUSEND y TURCV a través del parámetro ADDR. Los datos se guardan como el tipo de datos del PLC TADDR_Param que contiene una matriz para la dirección IP y el número de puerto. El controlador del robot envía los mensajes del robot al Nodos UDP a través del puerto 9050_{DIC}. Para el controlador de robot KUKA, se puede utilizar la misma variable para el parámetro ADDR para los bloques TUSEND y TURCV, ya que ambos Los comandos y los mensajes se comunican a través del mismo puerto.

Envío de comandos

A través del bloque TUSEND, los comandos deseados se envían al controlador del robot con un flanco ascendente en la entrada REQ. La entrada LEN debe proporcionarse con el número de caracteres que se enviarán. El área de la memoria de origen donde se envía el comando almacenados deben transferirse en la entrada de DATOS.

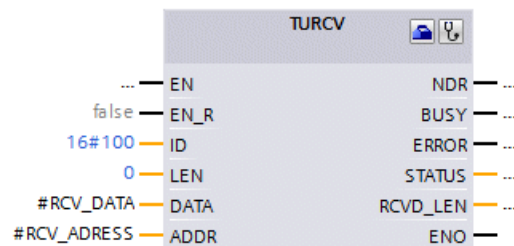
Figura 5-2 Bloque de función TU_SEND para enviar a través de una conexión UDP



Recibir mensajes de robot

Los mensajes del robot se reciben a través del bloque TURCV siempre que la entrada EN_R esté establecida en Verdadero. Al recibir mensajes de robot, nunca es necesario cambiar el valor 0 de la entrada LEN. El área de datos de destino en la que se guardará el mensaje se transfiere en la entrada de DATOS.

Figura 5-3 Bloque de función TU_RCV para recibir a través de una conexión UDP



A través de la salida NDR, el bloque envía un mensaje nuevo y, a través de la salida RCVD_LEN, transfiere el número de caracteres realmente recibidos al programa de usuario.

6 Apéndice

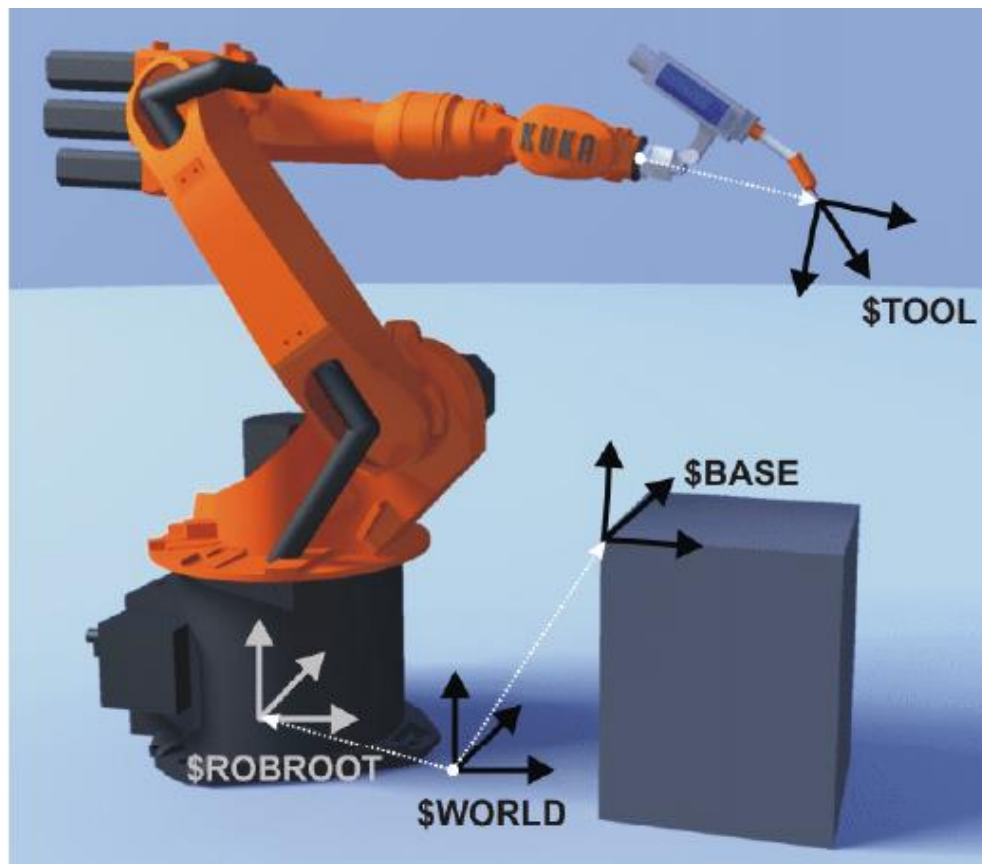
El objetivo de este capítulo es proporcionar información adicional sobre el manejo y la programación de robots industriales.

6.1 Sistemas coordinados

6.1.1 Visión general

Se definen varios sistemas de coordenadas en el controlador del robot. Determinan la posición de la pieza de trabajo y la herramienta y forman la base de los movimientos individuales del robot.

Figura 6-1 Vista general de los sistemas de coordenadas del robot



6.1.2 Sistema de coordenadas original del sistema (MUNDO)

El sistema de coordenadas WORLD es el sistema de coordenadas original de todo el sistema. Este sistema de coordenadas forma la base de todos los demás sistemas de coordenadas que resultan de la traslación y la rotación.

Si no se definen rotaciones y traslaciones de los otros sistemas de coordenadas, El origen del sistema de coordenadas WORLD se encuentra en la base del robot.

En el programa del robot, las rotaciones y traslaciones de los otros sistemas de coordenadas relativas al sistema de coordenadas WORLD se transfieren como una posición del robot con seis parámetros y los parámetros individuales tienen las siguientes funciones:

Coordenada X: traslación del sistema de coordenadas a lo largo del eje X.

Coordenada Y: traslación del sistema de coordenadas a lo largo del eje Y.

Coordenada Z: traslación del sistema de coordenadas a lo largo del eje Z.

Coordenada A: Rotación a la derecha del sistema de coordenadas sobre el eje X del sistema de coordenadas WORLD.

Coordenada B: Rotación a la derecha del sistema de coordenadas sobre el eje Y del sistema de coordenadas WORLD.

Coordenada C: Rotación a la derecha del sistema de coordenadas sobre el eje Z del sistema de coordenadas WORLD.

Nota

La rotación del sistema de coordenadas sobre los ejes del sistema de coordenadas WORLD sigue este orden:

Rotación alrededor del eje X por el valor absoluto de la coordenada A

Rotación alrededor del eje Y por el valor absoluto de la coordenada B

Rotación alrededor del eje Z por el valor absoluto de la coordenada C

6.1.3

Sistema de coordenadas de la base del robot (ROBROOT)

El sistema de coordenadas de la base del robot (ROBROOT) le permite lograr un desplazamiento del lugar de instalación del robot con respecto al origen del sistema de coordenadas WORLD.

Nota

Al iniciar el robot, el sistema de coordenadas de la base del robot debe definirse en los datos del controlador del robot.

Estos datos no se pueden modificar desde el programa de usuario mediante bloques de la librería de bloques KUKA.PLC mxAutomation.

6.1.4

Sistema de coordenadas de pieza (BASE)

El sistema de coordenadas de la pieza (BASE) se utiliza para definir el origen de la pieza a mecanizar en relación con el sistema de coordenadas WORLD.

La traslación y la rotación del sistema de coordenadas de la pieza (BASE) se pueden definir desde el programa de usuario utilizando la función KRC_WriteBaseData. La configuración actual del sistema de coordenadas o la traducción del sistema de coordenadas se pueden determinar utilizando la función KRC_ReadBaseData.

En el controlador del robot se pueden almacenar hasta 32 sistemas de coordenadas diferentes o traducciones de sistemas de coordenadas para el sistema de coordenadas de la pieza de trabajo (BASE). A continuación, se pueden seleccionar para definir fácilmente los comandos de movimiento del robot utilizando el programa de usuario.

6.1.5

Sistema de coordenadas de herramienta (HERRAMIENTA)

El sistema de coordenadas de la herramienta (TOOL) se utiliza para definir el origen de la herramienta adjunta al robot en relación con el sistema de coordenadas WORLD.

La traslación y la rotación del sistema de coordenadas de la herramienta (TOOL) se pueden definir desde el programa de usuario utilizando la función KRC_WriteToolData. La configuración actual del sistema de coordenadas o la traducción del sistema de coordenadas se pueden determinar utilizando la función KRC_ReadToolData.

Se pueden almacenar hasta 16 herramientas diferentes o traducciones del sistema de coordenadas de herramientas (HERRAMIENTA) en el controlador del robot. A continuación, se pueden seleccionar para definir fácilmente los comandos de movimiento del robot mediante el programa de usuario.

6.2

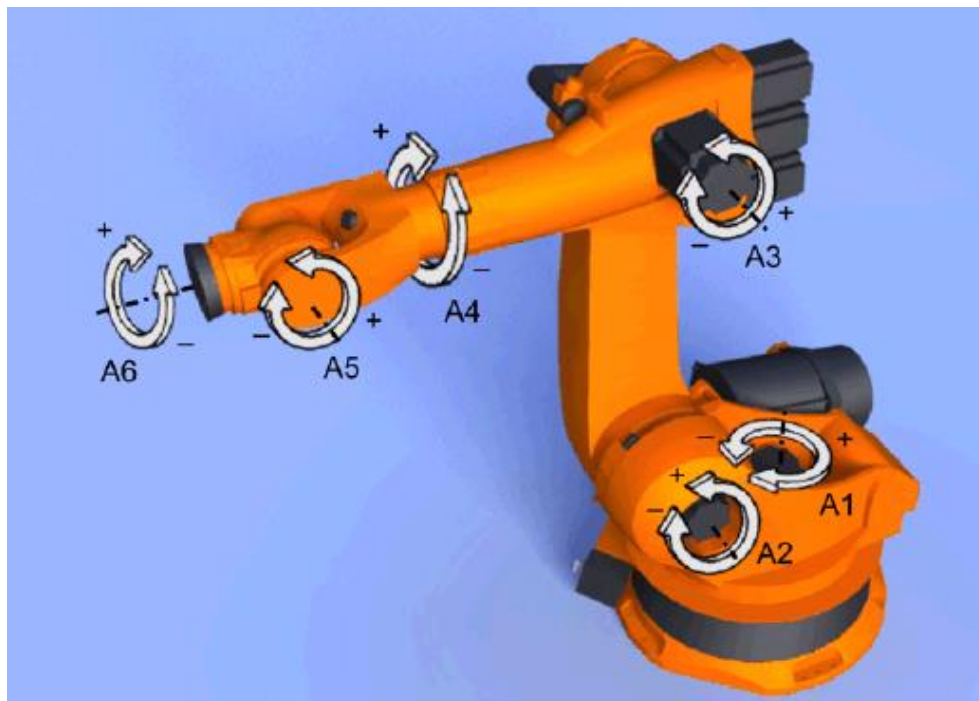
Definición de coordenadas

6.2.1

Ejes de robot

Un robot industrial de KUKA cuenta con hasta seis ejes individuales que permiten al usuario mover la cinemática del robot. Para posicionar el robot, estos ejes se pueden controlar directamente desde el programa de usuario utilizando las coordenadas del eje A1 a A6 o mediante la transformación de coordenadas en el controlador del robot especificando, desde el programa del usuario, una posición en un sistema de coordenadas definido en el controlador del robot, a través de las coordenadas cartesianas X, Y y Z y los ángulos de rotación alrededor de la coordenada ejes A, B y C.

Figura 6-2 Ejes de un robot de 6 ejes



Nota

Además de los ejes del robot A1 a A6, el controlador de robot KUKA proporciona seis ejes externos E1 a E6 que también pueden ser influenciados desde el controlador SIMATIC mediante el programa de usuario.

Sin embargo, los ejes externos (E1 a E6) no son el tema de este ejemplo. Para obtener más información, consulte la documentación de KUKA.PLC mxAutomation o la documentación del controlador del robot.

6.2.2 Coordenadas del eje

El posicionamiento directo de los ejes del robot se puede lograr especificando las posiciones de eje deseadas en el programa de usuario utilizando el tipo de datos E6AXIS. El

El comando KRC_MoveAxisAbsolute está disponible para este propósito.

Este comando mueve directamente los ejes del robot desde la posición actual a la posición especificada usando E6AXIS. En este proceso, no hay coordinación de los movimientos de los ejes individuales. Cada eje del robot se mueve a la posición especificada en el porcentaje especificado de su velocidad máxima de eje.

6.2.3 Coordenadas cartesianas

El tipo de datos E6POS y varios comandos de movimiento están disponibles para posicionar los ejes del robot con la ayuda de la transformación de coordenadas del controlador del robot.

Dependiendo del sistema de coordenadas seleccionado, como el sistema de coordenadas de la pieza (BASE) o el sistema de coordenadas de la herramienta (TOOL), los ejes del robot se pueden posicionar especificando una posición cartesiana y el vector de orientación en esta posición a través de la transformación de coordenadas.

Dependiendo del comando de movimiento seleccionado, la transformación de coordenadas del controlador del robot coordina los ejes del robot, por ejemplo, para el movimiento del punto de referencia del robot en una línea recta o una trayectoria circular.

6.2.4 Datos adicionales: "Estado" y "Turno"

Si el robot se posiciona mediante la transformación de coordenadas del controlador del robot, especificar la posición (X, Y y Z) y la orientación (A, B y C) no es suficiente para definir de manera inequívoca la posición del eje del robot en cada posición. Los datos adicionales para Status (S) y Turn (T) están disponibles para este propósito.

Nota

El controlador del robot considera los valores de estado y giro especificados solo para movimientos de punto a punto (PTP). Por lo tanto, el primer movimiento en un programa de movimiento del robot siempre debe ser un movimiento PTP completamente definido para que se defina una posición de inicio única para el robot. Para todos los demás comandos de movimiento, no es necesario cambiar el valor 0 de Status y Turn. El propio controlador del robot define el eje posición.

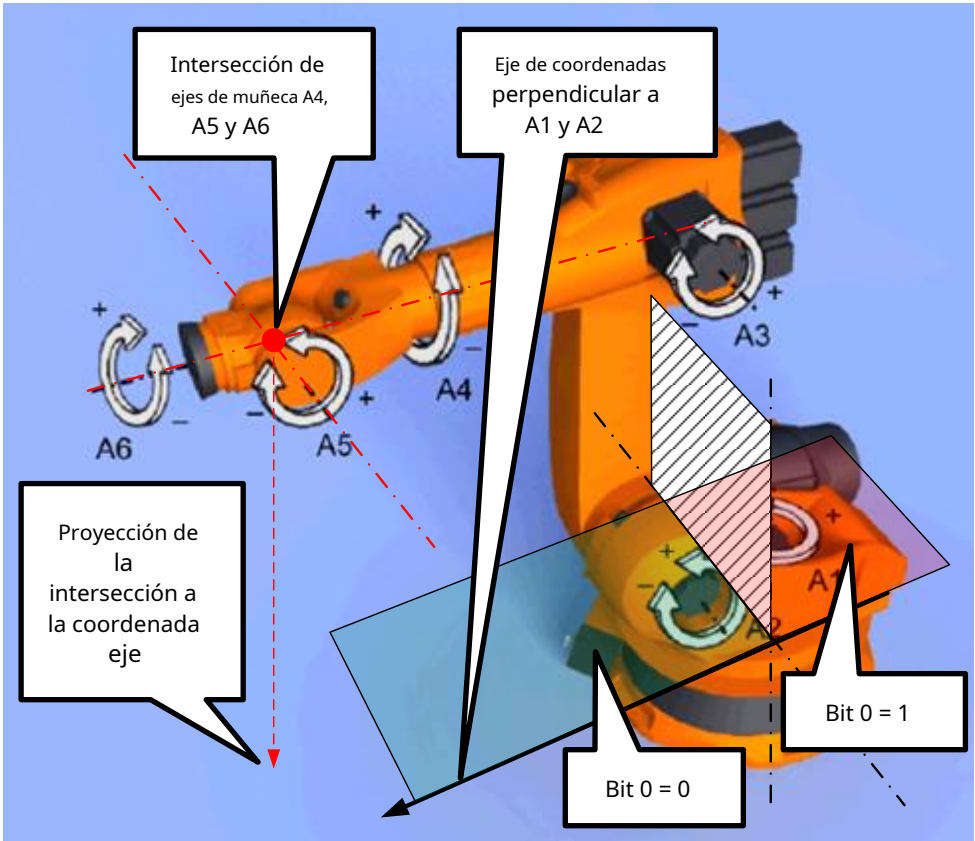
Datos adicionales: Estado

Los bits 0 a 4 de Estado le permiten definir la posición de los ejes del robot en relación con el sistema de coordenadas en el eje A1 en la base del robot.

Tabla 6-1 Datos adicionales: estado

Estado	Estado de bit	
	0	1
Bit 0	Área básica La intersección de los ejes de muñeca A4, A5 y A6 está por encima del <u>positivo</u> Eje X del sistema de coordenadas relativo al eje A1.	Área de arriba La intersección de los ejes de muñeca A4, A5 y A6 está por encima del <u>negativo</u> Eje X del sistema de coordenadas relativo al eje A1.
Bit 1	La posición del eje A3 es menor que cero.	La posición del eje A3 es mayor o igual a cero.
Bit 2	La posición del eje A5 es menor o igual a cero.	La posición del eje AA5 es mayor que cero.
Bit 3	No usado	No usado
Poco 4	El punto era <u>no</u> enseñado con un robot con una calibración absolutamente precisa.	El punto se enseñó con un robot con una calibración absolutamente precisa.
Poco 5	No usado	No usado
Poco 6	No usado	No usado
Poco 7	No usado	No usado

Figura 6-3 Ejemplo de datos adicionales de estado - bit 0



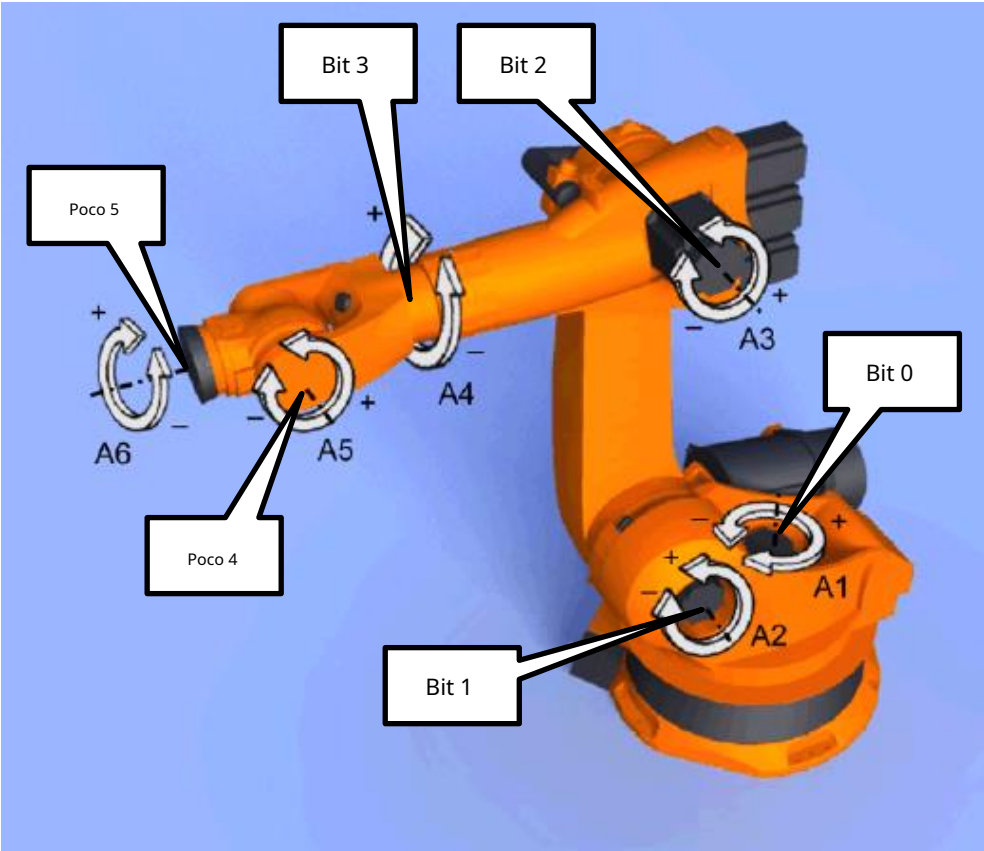
Datos adicionales: Turn

Los bits 0 a 5 de Turn le permiten definir la posición directa de los ejes del robot en el rangos de movimiento de los ejes.

Tabla 6-2 Datos adicionales: Turn

Girar	Estado de bit	
	0	1
Bit 0	La posición del eje A1 es mayor o igual a cero.	La posición del eje A1 es menor que cero.
Bit 1	La posición del eje A2 es mayor o igual a cero.	La posición del eje A2 es menor que cero.
Bit 2	La posición del eje A3 es mayor o igual a cero.	La posición del eje A3 es menor que cero.
Bit 3	La posición del eje A4 es mayor o igual a cero.	La posición del eje A4 es menor que cero.
Poco 4	La posición del eje A5 es mayor o igual a cero.	La posición del eje A5 es menor que cero.
Poco 5	La posición del eje A6 es mayor o igual a cero.	La posición del eje A6 es menor que cero.
Poco 6	No usado	No usado
Poco 7	No usado	No usado

Figura 6-4 Ejemplo de datos adicionales Turn



Nota

Si el estado y el giro no se especifican al posicionar el robot, el controlador del robot decide la transición de una posición de eje del robot a la siguiente.

6.2.5 Singularidades

Los robots industriales de seis ejes de KUKA tienen las siguientes tres singularidades diferentes:

Singularidad aérea

El eje A5 es perpendicular al eje A1.

Singularidad de posición extendida

El eje A5 está en la extensión de los ejes del robot A2 y A3.

Singularidad del eje de la muñeca

Los ejes A4 y A6 son paralelos entre sí y el eje A5 se mueve a la posición extendida A5, es decir, el eje A5 está en el rango de $\pm 0,01812^\circ$.

Al programar o acercarse a puntos manualmente, considere y, cuando sea posible, evite estas singularidades o utilice una programación diferente - por ejemplo, mueva directamente los ejes del robot en lugar de moverlos usando un comando de movimiento cartesiano - para evitarlos.

AVISO

Si hay al menos una singularidad en el espacio de trabajo deseado del robot, puede ser necesario verificar y corregir la posición del robot en el sistema para que la singularidad ya no ocurra en el espacio de trabajo.

6.2.6

Punto central de la herramienta (TCP)

Todos los comandos de movimiento cartesiano del robot se refieren al punto central de la herramienta (TCP). El punto de referencia de una herramienta se define escribiendo un corrector de herramienta relativo al sistema de coordenadas WORLD usando el comando KRC_WriteToolData.

Si no se selecciona ninguna herramienta, el TCP está en la brida del robot, es decir, una herramienta sin traslación y se asume la rotación.

7 Enlaces y literatura

Tabla 7-1

	Tema	Título
\1\	Industria Siemens Soporte en línea	http://support.industry.siemens.com
\2\	Descargar página de la entrada	https://support.industry.siemens.com/cs/ww/en/view/109482123
Siemens SIMATIC S7 / Safety Integrated / PROFIenergy		
\3\	SIMATIC PASO 7 Profesional	SIMATIC Manual del sistema STEP 7 Professional V13 SP1 Edición: 12/2014 ID del documento: Impresión de la ayuda en línea Número de pedido: - https://support.industry.siemens.com/cs/ww/en/view/109011420/
\4\	SIMATIC PROFINET	Descripción del sistema SIMATIC PROFINET Manual del sistema Edición: 03/2012 ID de documento: A5E00298287-06 Número de pedido: - https://support.industry.siemens.com/cs/ww/en/view/19292127/
\5\	SIMATIC La seguridad	Software industrial SIMATIC SIMATIC Safety - Configuración y programación Manual de programación y funcionamiento Edición: 11/2014 ID de documento: A5E02714439-AD Número de pedido: - https://support.industry.siemens.com/cs/ww/en/view/54110126
\6\	PROFIenergía	Guía de aplicación para implementar los conceptos de apagado con PROFIenergy Ejemplo de aplicación Edición: 07/2014 https://support.industry.siemens.com/cs/ww/en/view/96837137
\7\	PROFIenergía	¿Qué módulos de la cartera de Siemens son compatibles con las funciones de PROFIenergy? <small>Preguntas más frecuentes</small> Edición: 08/2014 https://support.industry.siemens.com/cs/ww/en/view/95848378
\8\	PROFIenergía	SIMATIC S7 Biblioteca para la configuración simple de PROFIenergy Ejemplo de aplicación / Biblioteca de bloques Edición: 07/2015 https://support.industry.siemens.com/cs/ww/en/view/109478388
KUKA PLC mx Automatización		
\9\	KUKA pagina de inicio	Robótica KUKA http://www.kuka-robotics.com Centro de descargas de KUKA: folletos http://www.kuka-robotics.com/germany/en/downloads/search/

	Tema	Título
\ 10 \	KUKA Software del sistema	KUKA Software del sistema Software del sistema KUKA - Versión 8.3 Instrucciones de operación y programación para integradores de sistemas Edición: 18/12/12 Versión: KSS 8.3 SI V1 en (PDF) Número de pedido: - Esta documentación se puede obtener en la línea directa de KUKA.
\ 11 \	KUKA Sistema Tecnología	Tecnología del sistema KUKA KUKA.PLC mxAutomation S7 - Versión 2.0 Descripción del producto y manual Edición: 05/03/15 Versión: S7_Library_mxAutomation_20_en.pdf Número de pedido: 00-233-923 Esta documentación se puede obtener en la línea directa de KUKA.
\ 12 \	KUKA Sistema Tecnología	Tecnología del sistema KUKA KUKA.SafeOperation - Versión 3.2 Instrucciones de montaje y funcionamiento Edición: 18/03/13 Versión: KST SafeOperation 3.2 V2 en Número de pedido: - Esta documentación se puede obtener en la línea directa de KUKA.
\ 13 \	KUKA Opción de controlador	Opción de controlador KUKA Documentación del controlador / dispositivo KUKA.ProfiNet 3.2 Edición: 03/09/14 Versión: KUKA.ProfiNet 3.2 V2 Número de pedido: - Esta documentación se puede obtener en la línea directa de KUKA.

8 Contacto

Si tiene alguna pregunta sobre los robots industriales KUKA o la biblioteca de bloques KUKA.PLC mxAutomation o desea solicitar la documentación necesaria de KUKA, utilice los datos de contacto que se indican a continuación.

8.1 Soporte de Siemens Robotics

Para ponerse en contacto con el soporte de SIEMENS Robotics, utilice la siguiente dirección de correo electrónico:

Correo electrónico: robotic.industry@siemens.com

8.2 KUKA Roboter GmbH

KUKA Roboter GmbH - Sede central
Zugspitzstraße 140
86165 Augsburg, Alemania
Teléfono: +49821797-4000
Fax: +49821797-4040

Correo electrónico: info@kuka-roboter.de



8.3 Línea directa de KUKA Roboter GmbH

KUKA Roboter GmbH - Línea directa
En Alemania, la línea directa está disponible las 24 horas del día
y los 365 días del año:
Teléfono: +49821797-1926



8.4 Ventas de automatización de KUKA.PLC mx

KUKA Roboter GmbH - Centro de ventas global
Hery-Park 3000
86368 Gersthofen, Alemania
Teléfono: +49821 4533-0
Fax: +49821 4533-1616
Correo electrónico: sales@kuka-roboter.de



Contacto:

Philipp Kremer
Gerente de producto
Teléfono: +49821 4533-3734
Fax: +49821797-41 3734
Philipp.Kremer@kuka.com



9 Historia

Tabla 9-1

Versión	Fecha	Modificaciones
V1.0	01/2016	Primera versión
V1.1	01/2016	Texto agregado / modificado